

**VŠB – Technická univerzita Ostrava**  
**Fakulta elektrotechniky a informatiky**  
**Katedra kybernetiky**  
**a biomedicínského inženýrství**

**Vzdálené monitorování a řízení**  
**technologických procesů**  
Remote Technological Process  
Monitoring and Control

## Zadání diplomové práce

Student: **Bc. Tomáš Hranický**  
Studijní program: N2649 Elektrotechnika  
Studijní obor: 2601T004 Měřicí a řídicí technika  
Téma: **Vzdálené monitorování a řízení technologických procesů**  
**Remote Technological Process Monitoring and Control**

### Zásady pro vypracování:

Diplomová práce se zabývá realizací vzdáleně říditelné a monitorovatelné laboratorní úlohy. Úkolem je především provedení funkčního propojení a zabezpečení jednotlivých komponent, kamery pro přenos obrazu a uživatelského rozhraní pro parametrizaci a spuštění běhu úlohy.

V souhrnu je práce charakterizována těmito body:

1. Návrh topologie a řešení pro vybranou úlohu.
2. Návrh a řešení řízení úlohy pomocí webové aplikace.
3. Návrh a řešení monitorování provozu úlohy.
4. Návrh a řešení zabezpečení realizovaného řešení.
5. Test celého systému a zhodnocení dosažených výsledků.

### Seznam doporučené odborné literatury:

- [1] LIPIANSKY, Ed. *Embedded Systems Hardware for Software Engineers*. 1. vyd. New York: McGraw-Hill Companies, 2011. 320 s. ISBN 978-0071639484.  
[2] POWERS, David. *PHP Solutions: Dynamic Web Design Made Easy*. 2. vyd. New York: Friends of ED, 2010. 528 s. ISBN 978-1430232490.  
[3] KAMAL, Raj. *Embedded Systems: Architecture, Programming and Design*. Vyd. 2. NewDelphi(India):Tata McGraw-Hill Publishing Company, 2009. 704 s. ISBN 978-0-07-015125-3.

Formální náležitosti a rozsah diplomové práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí diplomové práce: **Ing. Zdeněk Slanina, Ph.D.**

Datum zadání: 16.11.2012

Datum odevzdání: 07.05.2013



doc. Ing. Jiří Koziolek, Ph.D.  
vedoucí katedry

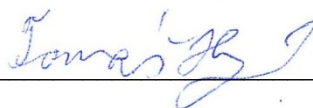


prof. RNDr. Václav Snášel, CSc.  
děkan fakulty

## **Prohlášení studenta**

„Prohlašuji, že jsem tuto práci vypracoval samostatně a uvedl jsem všechny literární prameny, publikace a zdroje, z nichž jsem čerpal.“

V Ostravě dne 7. května 2013



---

## **Poděkování**

Je mou milou povinností poděkovat všem, kteří mi svými cennými radami a vlastními zkušenostmi pomohli dovést práci ke zdárnému konci, a neméně také těm, již s pochopením akceptovali mé mnohdy zdánlivě nekonečné trávení času nad touto prací, především své rodině a blízkým.

Tomáš Hranický

## **Abstrakt**

Cílem této práce je vývoj systému vzdáleného monitorovacího a řídicího systému technologických procesů obsahující výběr komunikačního rozhraní, vývoj elektronických ovládacích obvodů, vývoj ovládacího softwaru a grafického webového uživatelského rozhraní.

Systém vzdáleného řízení a monitorování je aplikován na laboratorní úlohu měření na modelu kapacitního snímače.

Práce popisuje možné varianty řešení v oblasti výběru řídicí jednotky systému, následně popis realizačního postupu, funkcí systému a dosažených výsledků.

Ve vybraném řešení je použit minipočítač Raspberry Pi spolupracující s ovládacími elektronickými obvody, které dále obsahují síťový napájecí zdroj, řídicí obvody krokového motoru s mikrokontrolérem PIC, reléové spínače řízené logickými obvody řady 74HCxx. V oblasti softwaru jsou použity technologie HTML, PHP, SVG, programovací jazyk C# a operační systém Linux.

## **Abstract**

The target of this diploma work is development of remote technological process monitoring and control system. It includes choice of communication interface, development of electronic control circuit, development of control software and web user interface.

System has been applied to laboratory measurement of capacity sensor parameters.

There have been described more alternatives of solution, procedure of realization and final function of whole system in this work.

There has been used number of technologies in selected solution: Minicomputer Raspberry Pi, electronic circuits for power supply, stepper motor driver with PIC microcontroller, relay switches controlled by logical circuit 74HCxx series. In software there have been used HTML, PHP, SVG, C# programming language and operating system Linux.

## **Klíčová slova**

kapacitní snímač, krokový motor, měření kapacity, mikrokontrolér PIC, monitorovací systém, Raspberry Pi, řídicí systém, vzdálené řízení, webová vizualizace

## **Key words**

capacity measurement, capacity sensor, control system, monitoring system, PIC microcontroller, Raspberry Pi, stepper motor, web visualization

## Seznam použitých symbolů a zkratek

<b>ASCII</b>	American Standard Code for Information Interchange (americký standardní kód pro výměnu informací)
<b>FTP</b>	File Transfer Protocol (protokol pro přenos souborů)
<b>GPIO</b>	General Purpose Input / Output (univerzální vstup / výstup)
<b>HDMI</b>	High-Definition Multimedia Interface (rozhraní pro přenos médií ve vysokém rozlišení)
<b>HTML</b>	HyperText Markup Language (hypertextový značkovací jazyk)
<b>I<sup>2</sup>C</b>	Inter-Integrated Circuit (sběrnice pro komunikaci mezi integrovanými obvody)
<b>IP</b>	Internet Protocol (internetový protokol)
<b>JSA</b>	Jazyk symbolických adres
<b>LAN</b>	Local Area Network (lokální síť)
<b>LED</b>	Light-emitting diode (svítivá dioda)
<b>MD5</b>	Message-Digest algorithm
<b>MOSFET</b>	Metal Oxide Semiconductor Field Effect Transistor (tranzistor řízený polem s hradlem realizovaným vrstvou kovu odděleným od polovodiče dielektrickou vrstvou oxidu daného polovodiče) [18]
<b>PC</b>	Personal Computer (osobní počítač)
<b>PHP</b>	Personal Home Page
<b>PLC</b>	Programmable Logic Controller (programovatelný logický automat)
<b>RAM</b>	Random-access memory (paměť s přímým přístupem)
<b>SPI</b>	Serial Peripheral Interface (sériové rozhraní pro periferie)
<b>SQL</b>	Structured Query Language (strukturovaný dotazovací jazyk)
<b>SSH</b>	Secure shell
<b>SVG</b>	Scalable Vector Graphics (škálovatelná vektorová grafika)
<b>UART</b>	Universal Synchronous / Asynchronous Receiver and Transmitter (univerzální synchronní / asynchronní přijímač a vysílač)
<b>USB</b>	Universal Serial Bus (univerzální sériová sběrnice)
<b>VŠB – TUO</b>	Vysoká škola báňská – Technická univerzita Ostrava

# Obsah

<b>1</b>	<b>ÚVOD .....</b>	<b>1</b>
<b>2</b>	<b>TEORETICKÝ ROZBOR .....</b>	<b>2</b>
2.1	OBEČNÉ POŽADAVKY NA ČINNOST ZAŘÍZENÍ .....	2
2.2	VÝBĚR TYPU VZDÁLENÉ STANICE A KOMUNIKAČNÍHO ROZHRANÍ .....	3
2.3	NÁVRHY MOŽNÝCH ŘEŠENÍ .....	3
2.4	VÝBĚR A ROZVAHA VHODNÉHO ŘEŠENÍ .....	6
2.4.1	<i>Technický popis Raspberry Pi .....</i>	<i>6</i>
2.4.2	<i>Rozvaha konstrukčního řešení vyvíjeného přípravku .....</i>	<i>7</i>
<b>3</b>	<b>VÝVOJ ZÁKLADNÍHO KOMUNIKAČNÍHO SYSTÉMU MEZI PŘÍPRAVKEM A VZDÁLENOU STANICÍ .....</b>	<b>8</b>
3.1	INSTALACE PROGRAMOVÉHO VYBAVENÍ NA RASPBERRY PI .....	8
3.1.1	<i>Rozvaha programového řešení .....</i>	<i>9</i>
3.1.2	<i>Výběr a instalace webového serveru .....</i>	<i>9</i>
3.1.3	<i>Řešení vizuálního dohledu nad technologickým procesem .....</i>	<i>10</i>
3.1.4	<i>Umístění zkušební stránky na webový server .....</i>	<i>11</i>
<b>4</b>	<b>VÝBĚR TECHNOLOGICKÉHO PROCESU K ŘÍZENÍ A MONITOROVÁNÍ .....</b>	<b>12</b>
4.1	TEORETICKÝ ROZBOR – DIELEKTRICKÉ OBVODY .....	12
4.2	STÁVAJÍCÍ TECHNICKÉ PROVEDENÍ LABORATORNÍ ÚLOHY .....	15
4.2.1	<i>Nedostatky stávajícího řešení .....</i>	<i>16</i>
<b>5</b>	<b>BLOKOVÝ NÁVRH TOPOLOGIE A KONSTRUKČNÍHO ŘEŠENÍ SYSTÉMU .....</b>	<b>17</b>
5.1	NÁVRH NOVÉHO KONSTRUKČNÍHO PROVEDENÍ ÚLOHY .....	17
5.1.1	<i>Návrh nového mechanického provedení kapacitního snímače .....</i>	<i>17</i>
5.1.2	<i>Návrh posuvného a pohonného mechanismu .....</i>	<i>19</i>
5.2	BLOKOVÝ NÁVRH ŘÍDICÍ ELEKTRONIKY .....	20
5.3	NÁVRH PROGRAMOVÉ TOPOLOGIE SYSTÉMU .....	21
<b>6</b>	<b>FUNKČNÍ POPIS ŘÍDICÍ ELEKTRONIKY SYSTÉMU .....</b>	<b>24</b>
6.1	FUNKČNÍ POPIS BLOKU NAPÁJECÍHO ZDROJE SYSTÉMU .....	24
6.1.1	<i>Napájecí zdroj – nízkonapěťová část .....</i>	<i>24</i>
6.1.2	<i>Napájecí zdroj – síťová část .....</i>	<i>27</i>
6.2	FUNKČNÍ POPIS BLOKU POHONNÉ JEDNOTKY MECHANISMU .....	28
6.3	FUNKČNÍ POPIS BLOKU OVLÁDÁNÍ SEGMENTOVANÉ ELEKTRODY .....	30

<b>7</b>	<b>FUNKČNÍ POPIS PROGRAMOVÉ ČÁSTI SYSTÉMU .....</b>	<b>32</b>
7.1	FUNKČNÍ POPIS PROGRAMU PRO MIKROKONTROLÉR V MODULU POHONNÉ JEDNOTKY .....	32
7.2	FUNKČNÍ POPIS WEBOVÉ APLIKACE VE SPOLUPRÁCI S DATABÁZÍ JAKO ULOŽIŠTĚM DAT .....	33
7.2.1	<i>Popis rozložení webového uživatelského rozhraní .....</i>	<i>33</i>
7.2.2	<i>Popis datových toků ve vztahu k webovému rozhraní.....</i>	<i>36</i>
7.3	FUNKČNÍ POPIS KONZOLOVÉ APLIKACE PRO RASPBERRY PI.....	37
<b>8</b>	<b>NÁVRH ŘEŠENÍ ZABEZPEČENÍ UŽIVATELSKÉHO PŘÍSTUPU.....</b>	<b>39</b>
<b>9</b>	<b>DOSAŽENÉ VÝSLEDKY .....</b>	<b>41</b>
9.1	POTENCIÁL DALŠÍHO ROZVOJE SYSTÉMU .....	41
<b>10</b>	<b>ZÁVĚR .....</b>	<b>43</b>
<b>11</b>	<b>POUŽITÁ A NASTUDOVANÁ LITERATURA.....</b>	<b>44</b>
<b>12</b>	<b>SEZNAM PŘÍLOH .....</b>	<b>46</b>

# 1 Úvod

Téma této práce, vzdálené monitorování a řízení technologických procesů, bylo zvoleno především z důvodu stále se zvyšující popularity tohoto technického odvětví. Jde o téma, které je součástí stále většího okruhu průmyslových oborů, vědecko-výzkumných aplikací i oborů lidské činnosti, které mnohdy ani nejsou ve svém základu s technikou přímo spjaty.

Vzdálené monitorování a řízení technologických procesů má již řádově desítky let své využití v lehkém i těžkém průmyslu, kde může přispívat ke zdokonalení koordinace jednotlivých částí a etap technologických procesů, k eliminaci chyb a zpoždění způsobovaných lidským faktorem, k rychlejší a přesnější detekci poruch a tím zefektivnění celé výroby. Úzce toto téma také souvisí s centralizací monitorování a řízení nejen technologických procesů do nadřízených dispečerských pracovišť. Příkladem centrálního vzdáleného řízení a dozoru mohou být nejen automatické linky ve výrobních halách, ale také dopravní dispečinky, kde dochází k on-line komunikaci mezi dispečerským softwarem a palubními počítači ve vozidlech. Dalším příkladem může být systém distribuce elektrické energie, kde je dispečerské řízení téměř nutností pro okamžité reakce na aktuální odběr ze sítě, včasné zjištění poruch a tím minimalizaci dob výpadků u koncových odběratelů.

Stejně jako je tomu u většiny produktů technického pokroku a vývoje, i vzdálené monitorování a řízení po tom, co bylo vyvinuto a praktikováno v průmyslu, dostává se v současné době do mnoha aplikací v oblasti komerční elektroniky. Pravděpodobně nejdéle a nejhojněji je využíváno v zabezpečovací technice. Další aplikace těchto technologií jsou v současné době ve vývoji – na zahraničních trzích se začínají objevovat domácí spotřebiče s možností vzdáleného monitorování případně řízení, např. automatická pračka s Wi-Fi komunikačním rozhraním. Dále je očekáván rozvoj tzv. inteligentních domů, kde jsou centralizovány programovatelné systémy řízení osvětlení, temperování, zabezpečení domu apod.

Jak vyplývá z předchozích odstavců, zvolené téma je již v mnoha aplikacích uspokojivě vyřešeno, v některých je dokonce nutností. Výsledkem této práce by však mělo být konkrétní řešení navržené především pro méně náročné aplikace s vyšším zohledněním efektivity z ekonomického hlediska, jednoduchosti a v neposlední řadě také určité univerzality řešení v daném okruhu aplikací.



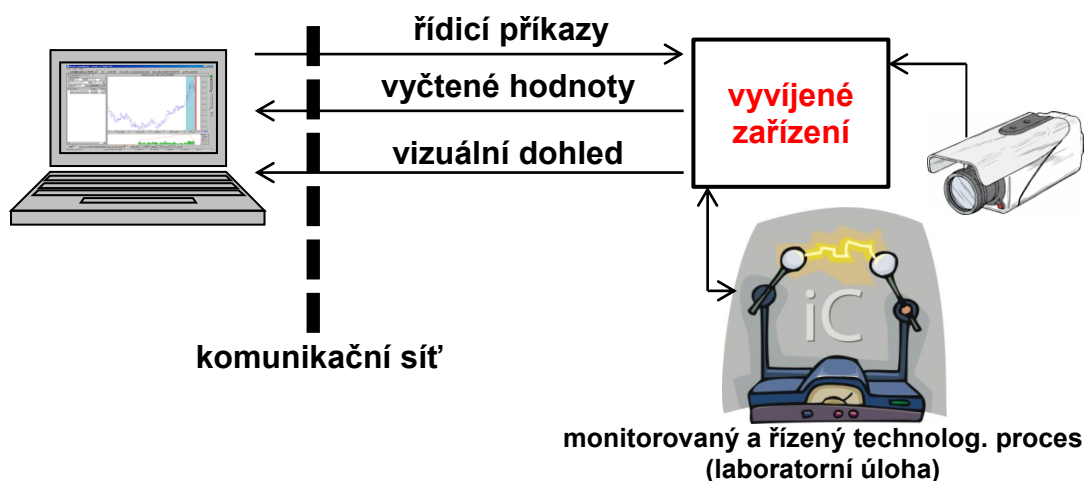
## 2 Teoretický rozbor

### 2.1 Obecné požadavky na činnost zařízení

Cílem práce je návrh a konstrukce přípravku, pomocí kterého bude možno ze vzdálené stanice monitorovat a řídit zvolené zařízení o rozsahu a složitosti běžné výukové laboratorní úlohy.

Navrhovaný přípravek musí zvládnout úkony znázorněné na obr. 1 – vyčíst, případně patřičně předzpracovat informaci dodanou sledovaným zařízením a odeslat tuto informaci uživateli na vzdálené stanoviště. Rovněž musí být přípravek schopen přijmout instrukci odeslanou vzdáleným uživatelem, případně ji předzpracovat a předat ovládanému zařízení. V případě, že bude některým z možných technických řešení podporována spolupráce s videozařízením, bude řízené zařízení také vizuálně sledováno jednou či více kamerami a jejich obraz bude přenášen vzdálenému uživateli.

Zařízení pro realizaci vzdálené stanice a komunikační rozhraní musí být zvoleny tak, aby jej bylo možno využít s minimálním ohledem na jeho umístění a vzdálenost od řízeného zařízení.



Obr. 1 – schéma základního principu funkce kompletního vyvíjeného systému.

## 2.2 Výběr typu vzdálené stanice a komunikačního rozhraní

Na místě vzdálené monitorovací a řídicí stanice je především z důvodu univerzality, tedy nezávislosti na jednom konkrétním účelovém výrobku, a také minimalizace výrobních nákladů uvažován běžný PC, notebook, tablet či jiné zařízení disponující srovnatelnými výkonnostními, zobrazovacími a softwarovými možnostmi.

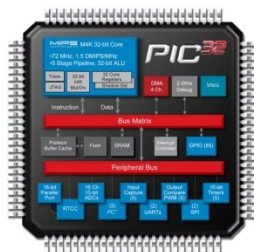
Aby byl splněn požadavek na libovolnost umístění vzdálené stanice, je nutno zvolit připojení vyvíjeného přípravku a vzdálené stanice do některé z globálních komunikačních sítí. Z těchto se nabízejí GPRS či síť internet. Vzhledem k tomu, že v tomto případě není shledáno nutným řešit připojení vzdálené stanice k zařízení z míst bez připojení k internetu, je opět především z ekonomických důvodů zvolena varianta připojení zařízení do sítě internet. Toto zjednodušuje situaci jak na straně ovládaného zařízení, tak na straně vzdálené stanice, jelikož síťovou kartou s ethernetovým rozhraním disponují prakticky všechny běžně vyráběné PC a notebooky.

## 2.3 Návrhy možných řešení

Výše uvedené požadavky je možno splnit za pomoci několika různých běžně dostupných prostředků. Jednotlivá řešení se pohybují v rozdílných cenových i výkonových relacích, jejich využití vyžaduje různé stupně programátorských znalostí. V jednotlivých dále uvedených řešeních jsou do různé míry integrována potřebná rozhraní a požadované funkce, a tím vyžadují různou míru dalších nezbytných hardwarových a softwarových úprav.

### a) Mikrokontrolér (obr. 2).

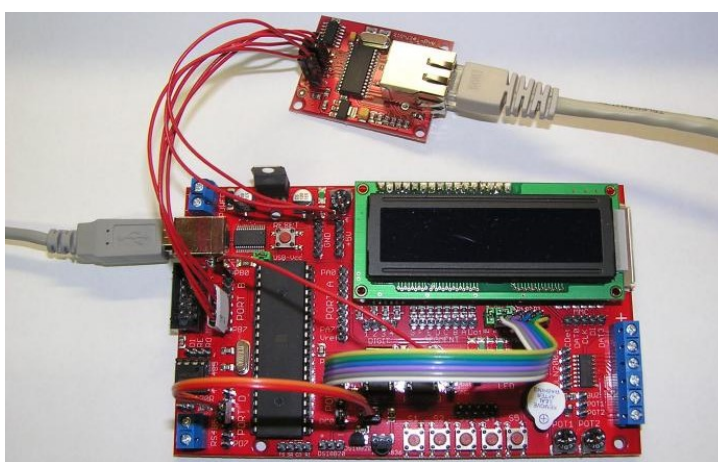
Jde jednoznačně o cenově nejpříjemnější řešení, cena běžných mikrokontrolérů se pohybuje v řádech desítek, nanejvýš stovek českých korun. Naproti nízké ceně by však řešení se samostatným mikrokontrolérem vyžadovalo poměrně velký počet externích součástek, vlastní elektrotechnické i mechanické zpracování a především velmi pokročilé znalosti v oboru programování dané rodiny mikrokontrolérů. Dále by v tomto případě byly pravděpodobně komplikované jakékoliv dodatečné úpravy softwaru zařízení např. pro pozměnění či rozšíření funkce přípravku, zejména pro jiného programátora, který není autorem původního softwaru. Nakonec je zde pochybnost, zda by běžný mikrokontrolér byl výkonově schopen zajistit přenos obrazu z kamery snímající ovládané zařízení.



Obr. 2 – příklad mikrokontroléru. [16]

b) Vývojový kit s mikroprocesorem či mikrokontrolérem (obr. 3).

Toto řešení je založeno na nástavbě prostředků využitých v navrhovaném řešení předchozím, kdy mikrokontrolér či mikroprocesor je integrován na vývojové desce již spolu s dalším hardwarem (převodníky, zobrazovací a ovládací prvky, komunikační rozhraní). Ve srovnání s předchozí možností zde téměř odpadá nutnost rozšíření přípravku o další součástky za předpokladu, že by byla vybrána vývojová deska s veškerým potřebným vstupně-výstupním hardwarem. Další nevýhody popsané v předchozí možnosti řešení však přetrvávají a navíc dochází v tomto případě k rapidnímu nárůstu ceny – až o jeden řád. Ceny vývojových kitů vhodných pro tuto aplikaci se pohybují od jednoho po několik tisíc českých korun.



*Obr. 3 – příklad vývojového kitu s LAN adaptérem. [7]*

c) Programovatelný automat (PLC), (obr. 4).

Další možností řešení je využití programovatelného automatu PLC. Tato koncepce řešení by znamenala minimalizaci nutného externího hardwaru, jelikož PLC nabízí analogové i digitální vstupy a výstupy, navíc v modulovém provedení jejich rozšiřitelný počet, který by byl pro danou aplikaci dostačující, a dále potřebné ethernetové komunikační rozhraní. Toto řešení již nabízí výrazné zjednodušení i z pohledu softwarového, protože s programovatelnými automaty je dodáván i již poměrně uživatelsky příjemné vývojové prostředí a mnohé PLC již disponují také vlastním webovým serverem, který může být v této aplikaci využit. Prvním negativem v případě využití PLC je nedostupnost jakéhokoliv portu, který by umožňoval připojení kamery pro přenos obrazu, druhým výraznějším negativem je pak velmi vysoká cena automatu, která se pohybuje v řádech tisíců až desítek tisíc českých korun.



Obr. 4 – PLC s LAN portem. [17]

d) PC či embedded computer s měřicí kartou (obr. 5).

Posledním obecně dostupným řešením je využití kompletního PC či vestavného systému s měřicí kartou. Jde o řešení, které díky měřicí kartě nevyžaduje mnoho dalších externích součástí a nabízí prakticky libovolné softwarové možnosti zpracování dat a instrukcí poměrně vysokým výpočetním výkonem ve srovnání se všemi předcházejícími možnostmi. Stejně tak je tímto řešením umožněno připojení libovolného vizuálního či dokonce akustického záznamového zařízení. Oproti předcházejícím možnostem má toto řešení nevýhodu v poměrně vysokém příkonu, velkých rozměrech a ceně srovnatelné s variantou c).



Obr. 5 – PCI měřicí karta. [14]

## 2.4 Výběr a rozvaha vhodného řešení

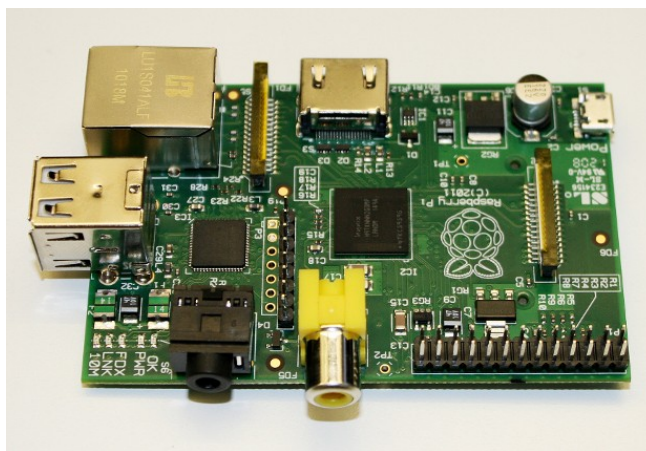
Je-li znovu vzato v úvahu zadání řešeného problému a požadované parametry výsledného řešení, tedy především nízká cena, jednoduchost řešení a v ideálním případě i kompaktní rozměry vyvíjeného přípravku, nelze označit za vyhovující ani jedno z výše popsaných technických řešení problému. Kromě těchto obecně známých a již několik let či desítek let na trhu dostupných řešení od různých výrobců a v mnoha modifikacích však existují také některé specifické typově ojedinělé byť sériově vyráběné produkty, které nebylo možno zařadit do žádné ze jmenovaných skupin výrobků. Jedním z nich je produkt Raspberry Pi britského výrobce Raspberry Pi Foundation.

### 2.4.1 Technický popis Raspberry Pi

Raspberry Pi je, jak uvádí výrobce, minipočítač o velikosti platební karty zobrazený na obr. 6. Jde o jednodeskový počítač založený na procesoru ARM s taktem 700 MHz, dále osazený grafickým procesorem, pamětí RAM 128, 256 nebo 512 MB (dle modelu), USB rozhraním, ethernetovým komunikačním rozhraním, video výstupy pro připojení monitoru a zvukovým výstupem, což jsou periférie, kterými disponuje běžný PC. Jako programové i datové uložení slouží SD karta vložená do integrovaného slotu. Navíc oproti běžnému PC je na desce k dispozici 8 GPIO pinů, UART, I<sup>2</sup>C a SPI sběrnice.

Pro běh na Raspberry Pi je určena speciální distribuce operačního systému Linux nazvaná Raspbian založená na Debianu optimalizovaná pro výkon Raspberry Pi.

Cena takto vybaveného minipočítače Raspberry Pi se pohybuje okolo jednoho tisíce českých korun, což ve srovnání s výše uvedenými možnostmi řešení patří k těm příznivějším nabídkám. Softwarové vybavení je na úrovni běžného PC, čili plně dostačující. Po hardwarové stránce disponuje Raspberry Pi při velmi kompaktních rozměrech všemi potřebnými rozhraními včetně USB portu pro možné připojení webkamery a navíc přímo GPIO piny pro monitorování resp. řízení výstupů resp. vstupů ovládaného zařízení. Z těchto důvodů je Raspberry Pi vybrán jako nejvhodnější prostředek pro realizaci vyvíjeného přípravku.



Obr. 6 – minipočítač Raspberry Pi. [26]



## 2.4.2 Rozvaha konstrukčního řešení vyvíjeného přípravku

V této části práce prozatím není stanoveno konstrukční provedení ovládaného zařízení, jelikož zatím není vybrán technologický proces (úloha), který bude řízen, nicméně obecně existují dvě možnosti, jak řešit mechanické a elektrické spojení zařízení a přípravku zajišťujícího možnost jeho vzdáleného monitorování a řízení.

- a) Mechanická a elektrická integrace celého modulu Raspberry Pi do ovládaného zařízení tak, aby vše tvořilo jednotlivý kompaktní funkční celek.
- b) Uložení Raspberry Pi do vlastního boxu, takovým způsobem, aby byly uživatelsky přístupné veškeré konektory Raspberry Pi, a zajištění vlastního nezávislého napájecího zdroje přípravku.

Zvolena je druhá možnost, tedy oddělené řešení a to především z důvodu dodržení jisté univerzality přípravku, kdy by tento mohl být v případě potřeby jednoduše od řízeného zařízení odejmut za účelem úpravy, opravy či využití v jiné aplikaci.

Pro účely mechanického osazení do dané aplikace je v současné době distribuovaný model Raspberry Pi vybaven dvěma montážními otvory. Pro tento minipočítač jsou dokonce na trhu dostupné účelové plastové krabičky (obr. 7 a 8) s továrními otvory pro potřebné konektory.



Obr. 7 – příklad na trhu dostupného boxu pro Raspberry Pi. [6]



Obr. 8 – příklad na trhu dostupného boxu pro Raspberry Pi. [13]

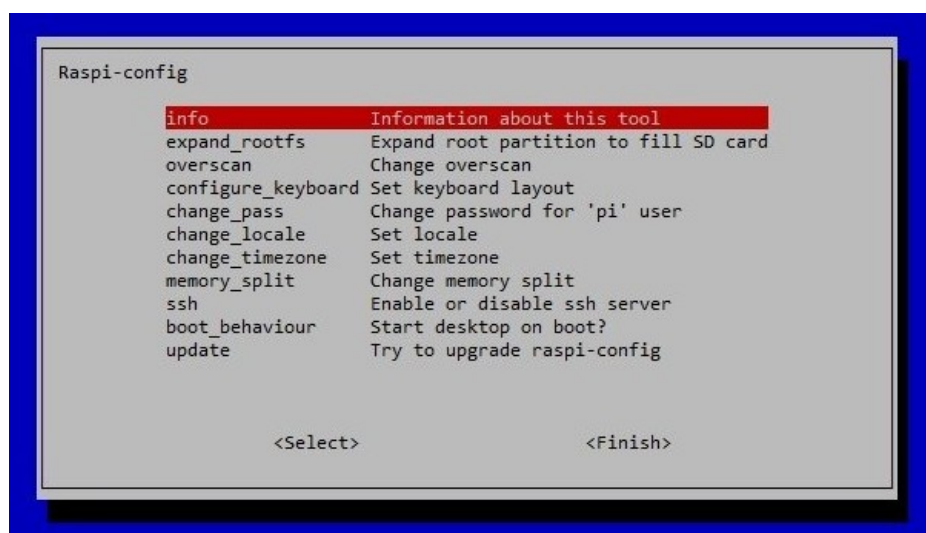
### 3 Vývoj základního komunikačního systému mezi přípravkem a vzdálenou stanicí

V této fázi je vybrána hardwarová technologie pro realizaci přípravku, a je nutno přistoupit k postupnému vývoji programové části komunikačního prostředí mezi přípravkem a vzdálenou stanicí. Pro potřeby vývoje a testování budoucích základních funkcí zařízení byly nejprve GPIO piny Raspberry Pi připojeny ke zkušební desce obsahující LED, které budou indikovat stav výstupních pinů, a testovací spínač, kterým budou nastavovány hodnoty na vstupních pinech.

Následně byl nainstalován a nastaven veškerý potřebný software na Raspberry Pi a vytvořeno testovací rozhraní pro stranu vzdáleného uživatele.

#### 3.1 Instalace programového vybavení na Raspberry Pi

Pro minipočítač Raspberry Pi je primárně určena distribuce Raspbian operačního systému Linux. Tato byla dle pokynů výrobce nahrána na SD paměťovou kartu, která nadále bude po celou dobu vložena ve slotu Raspberry Pi či v případě potřeby lze tuto paměťovou kartu kdykoliv použít v libovolném jiném kusu Raspberry Pi bez jakékoliv ztráty dat či funkčnosti. Před prvním připojením Raspberry Pi k napájecímu zdroji byly připojeny ještě klávesnice k USB portu a monitor k HDMI portu, případně myš ke druhému USB portu, pokud by uživatel dal přednost práci v grafickém prostředí operačního systému. Po prvním spuštění Raspberry Pi s nově nahanou paměťovou kartou, s čistou instalací operačního systému, je nejprve zobrazeno konfigurační menu (obr. 9) s možností nastavení základních parametrů. Poté, co jsou všechny tyto uživatelem nastaveny a uloženy, dojde ke spuštění operačního systému v textovém režimu, což pro tuto aplikaci postačuje. Dále je nutno zvolit způsob, jakým bude ze vzdálené stanice k přípravku přistupováno a provést případnou instalaci dalšího softwaru či změny konfigurace.



Obr. 9 – konfigurační menu Raspbianu při prvním spuštění operačního systému.

### 3.1.1 Rozvaha programového řešení

Jako pravděpodobně nejvhodnější řešení komunikace resp. vztahu mezi Raspberry Pi a vzdáleným počítačem byl zvolen vztah webový server a jeho klient. Webový server poběží na Raspberry Pi a vzdálený klient se k němu bude moci připojit za pomoci běžného webového prohlížeče, správce pak také za pomoci FTP či SSH klienta.

Výhoda tohoto systému je v první řadě nenutnost instalace jakéhokoliv speciálního účelového softwaru na počítač, který má být použit ve funkci vzdálené stanice. Další výhodou může být možnost připojení více uživatelů současně, při čemž samozřejmě musí být ošetřen přístup k zařízení tak, aby byl umožněn k řídicím funkcím vždy nejvýše jednomu uživateli, ale zároveň aby bylo ostatním uživatelům umožněno sledování systému.

### 3.1.2 Výběr a instalace webového serveru

Pro použití s Raspberry Pi byl zvolen nenáročný webový server Lighttpd. Tento byl nainstalován pomocí správce balíků do Raspbianu včetně PHP interpretu. Dále bylo nakonfigurováno síťové připojení a server byl spuštěn.

Raspberry Pi bude jak ve vývojové fázi, tak ve finálním provedení součástí lokální sítě, čili nebude mu přímo přidělena veřejná IP adresa. Konkrétně nyní v době vývoje zařízení je Raspberry připojen k veřejné síti skrze dva routery. Aby byl Raspberry Pi přístupný z veřejné sítě internet, je nutno na obou těchto routerech nastavit pro všechny potřebné porty tzv. IP forwarding (obr. 10). U některých typů routerů je funkce nazvána virtual server. Jde o nastavení definující, na které ze zařízení připojených k routeru budou předávány dotazy odesílané na danou veřejnou IP adresu, k níž je do veřejné sítě internet připojena celá daná lokální síť.

Port Range	Local IP	Local Port	Protocol	Protocol No.	Description
			TCP		
22	192.168.1.21	22	TCP		
80	192.168.1.21	80	TCP		
8081	192.168.1.21	8081	TCP		
8090	192.168.1.21	8090	TCP		
8082	192.168.1.21	8082	TCP		
3306	192.168.1.21	3306	TCP		

Obr. 10 – příklad vyplněné tabulky pro IP forwarding v konfiguraci routeru.



### 3.1.3 Řešení vizuálního dohledu nad technologickým procesem

Pro zajištění vizuálního dohledu nad řízeným technologickým procesem bude použita jedna či dvě, v případě, že zpracování dvou videostreamů Raspberry Pi výkonově zvládne, webové kamery připojené do USB portu Raspberry Pi. Ke zpracování a distribuci obrazu z kamer je nutno použít speciální software – prozatím byl vybrán video server Motion. V případě, že se osvědčí, bude jeho využití součástí finálního provedení.

Balík Motion byl nainstalován pomocí správce balíků do Raspbianu. Následuje úprava parametrů videa a síťové komunikace v konfiguračním souboru serveru motion. V případě, že má server pracovat pouze s jedinou webkamerou, postačí nastavit parametry v globálním konfiguračním souboru `/etc/motion/motion.conf` (ukázka z konfiguračního souboru – kód 1). Pokud je vyžadováno zpracování obrazu ze dvou až čtyř videozařízení, je nutno v globálním konfiguračním souboru odkomentovat příslušný počet odkazů na jednotlivé konfigurační soubory pro každé videozařízení a parametry, jež je vyžadováno mít různé u jednotlivých videozařízení, nastavit samostatně.

```
...
#####
# Live Webcam Server
#####
# The mini-http server listens to this port for requests (default: 0 =
disabled)
webcam_port 8081

#####
# Capture device options
#####
# Videodevice to be used for capturing (default /dev/video0)
videodevice /dev/video0

# Should normally be set to 1 for video/TV cards, and 8 for USB cameras
input 8
...
```

*Kód 1 – ukázka z konfiguračního souboru serveru Motion.*

Klíčová nastavení, která obsahuje konfigurační soubor serveru Motion, jsou např. port pro odesílání zpracovaného obrazu klientovi nebo určení zdrojového videozařízení.

Po dokončení zmíněných nastavení může být Motion spuštěn.

### 3.1.4 Umístění zkušební stránky na webový server

Nyní je možno na webový server do adresáře `/var/www/` umístit zkušební webovou stránku s testovacím kódem, pomocí něžž bude moci být vyzkoušena jak obousměrná komunikace (ukázky PHP kódů 2 a 3) mezi uživatelem vzdálené stanice a GPIO porty, tak funkce přenosu obrazu z webkamery.

Zkušební stránka je rozdělena na dva rámy. První rám má nastaveno automatické obnovování obsahu, aby bylo zajištěno aktuální zobrazení stavu vstupního zkušebního pinu na Raspberry Pi. Dále jsou v něm umístěny checkboxy pro změnu stavu dvou výstupních pinů Raspberry Pi.

```
if ( $_GET["zel"] === "on" ) {  
    shell_exec( "echo \"1\" > /sys/class/gpio/gpio4/value" );  
}  
else {  
    shell_exec( "echo \"0\" > /sys/class/gpio/gpio4/value" );  
}
```

*Kód 2 – ukázka PHP kódu ve zkušební stránce nastavujícího stav výstupního pinu Raspberr Pi dle stavu checkboxu na stránce.*

```
if ( shell_exec("cat /sys/class/gpio/gpio22/value") == 1 ) {  
    echo "<FONT Color = 'red'>Tlačítko stisknuto.</FONT>";  
}  
else {  
    echo "<FONT Color = 'green'>Tlačítko uvolněno.</FONT>";  
}
```

*Kód 3 – ukázka PHP kódu ve zkušební stránce zjišťujícího stav vstupního pinu Raspberr Pi.*

Druhý rám zkušební stránky obsahuje pouze obraz zaznamenávaný webkamerou. Obraz je na zdrojovém portu obnovován automaticky.

Zdrojovým obsahem druhého rámu jsou čistě data na portu určeném v konfiguraci serveru Motion, implicitně portu 8081.

Tímto je základní princip činnosti vyvíjeného přípravku otestována a může být přikročeno k vývoji s v konkrétní aplikaci.

## 4 Výběr technologického procesu k řízení a monitorování

Jako vhodný technologický proces pro aplikaci systému vzdáleného řízení a monitorování byla vybrána úloha „Kapacitní snímače“ z předmětu „Senzory a měření“ prováděná v laboratorních podmínkách studenty. Technické řešení navrhované v této práci primárně umožní provedení bodu 1. zadání úlohy „Změřte statickou charakteristiku modelu kapacitního snímače s proměnnou plochou překrytí pro různé tvary elektrod.“ Úplné zadání a popis úlohy jsou přiloženy na DVD s touto prací (více v příloze D).

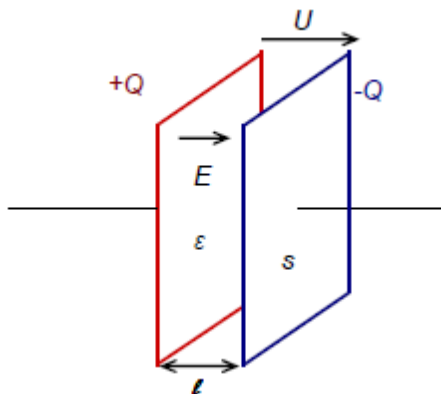
### 4.1 Teoretický rozbor – dielektrické obvody

Vybraná laboratorní úloha je založená na měření kapacity modelu kapacitního snímače s proměnnou plochou elektrod, případně proměnnou tloušťkou dielektrika a jeho materiálu.

#### Dielektrické obvody [5]

##### Kapacita C – jednoduchý dielektrický obvod

Mějme dvě rovnoběžně vodivé desky o ploše  $S$ , vzdálenost mezi nimi je  $l$  (obr. 11), prostor mezi nimi je vyplněn dielektrikem s permitivitou  $\varepsilon = \varepsilon_0 \cdot \varepsilon_r$  ( $\varepsilon_0 = 8,859 \cdot 10^{-12} \text{ F/m}$ ). Náboje na deskách jsou  $+Q$  a  $-Q$ .



Obr. 11 – model kondenzátoru.

Pokusy bylo zjištěno, že při vkládání různých materiálů (dielektrik) se napětí mezi deskami mění (vnější zdroj napětí není připojen, stále stejný náboj). Předpokládáme nejdříve vakuum (jeho  $\varepsilon_r = 1$ ), tedy  $\varepsilon = \varepsilon_0$ ; náboji  $Q$  odpovídá napětí  $U_0$ . V homogenním poli můžeme určit intenzitu elektrického pole  $E_0 = U_0 / l$ . Dále uvažujme obecné dielektrikum, jehož  $\varepsilon_r \geq 1$ ; stejnému náboji  $Q$  odpovídá nyní jiné napětí  $U$ . V homogenním poli můžeme určit novou

odpovídající intenzitu elektrického pole  $E = U/l$ . Právě tento poměr intenzit elektrického pole definuje relativní permitivitu dielektrika

$$\varepsilon_r = \frac{E_0}{E} = \frac{E_0 \cdot l}{E \cdot l} = \frac{U_0}{U}$$

Jestliže máme mezi vývody kondenzátoru (dielektrického obvodu) napětí  $U$  a známe i náboj  $Q$ , je tzv. kapacita (prostoru) definována poměrem

$$C = \frac{Q}{U} \quad [F]$$

nebo-li

$$Q = C \cdot U \quad [C]$$

náboj je přímo úměrný napětí a kapacitě obvodu. Odsud také plyne, že proud kapacitou je

$$i = \frac{dQ}{dt} = C \cdot \frac{du}{dt} \quad [A]$$

Z udělaných elementárních úvah je zřejmé, že při konstantním náboji se mění se záměnou dielektrika i napětí, proto se musí měnit i kapacita:

$$C_0 = \frac{Q}{U_0}$$

$$C = \frac{Q}{U} = \frac{Q}{\frac{U_0}{\varepsilon_r}} = \varepsilon_r \cdot \frac{Q}{U_0} = \varepsilon_r \cdot C_0$$

Pomocí Gaussovy věty lze určit, že intenzita elektrického pole  $E$  je vždy přímo úměrná velikosti náboje  $Q$ , nepřímo úměrná ploše  $s$ , na které vliv náboje zkoumáme (která náboj obklopuje) a materiálové konstantě (permitivitě)  $\varepsilon$ . V homogenním prostředí ( $l$  je podstatně kratší než další rozměry kondenzátoru) je elektrické pole soustředěno pouze do prostoru mezi deskami (s permitivitou  $\varepsilon$ ) – v dielektriku; mimo dielektrika je  $E = 0$ . Platí proto jednoduchý vztah

$$E = \frac{Q}{S \cdot \varepsilon} \quad \left[ \frac{V}{m} \right]$$

Současně ovšem v homogenním prostředí platí

$$U = E \cdot l \quad [V]$$

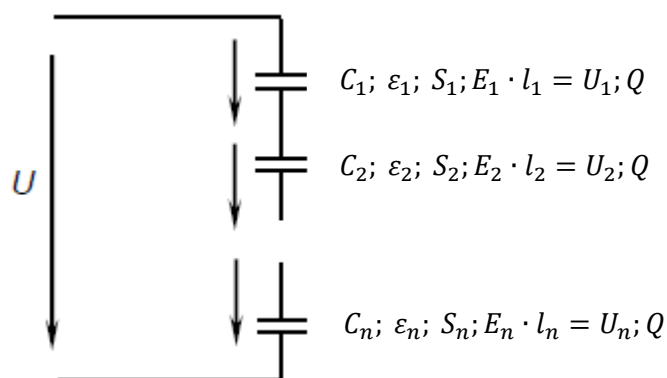
Snadno nyní určíme, že

$$Q = \varepsilon \cdot E \cdot S = \varepsilon \cdot S \cdot \frac{U}{l} = \frac{\varepsilon \cdot S}{l} \cdot U = C \cdot U$$

Je proto zřejmé, že

$$C = \frac{\varepsilon \cdot S}{l}$$

**Sériové řazení kapacit** (schéma na obr. 12)



Obr. 12 – sériové řazení kapacit.

Z fyzikální podstaty proudu je zřejmé, že sériovým řazením prochází stejný proud vždy po stejnou dobu, proto musí být ve všech kondenzátorech stejný náboj  $Q$ . Platí proto pro libovolné napětí

$$U_i = \frac{Q}{C_i}$$

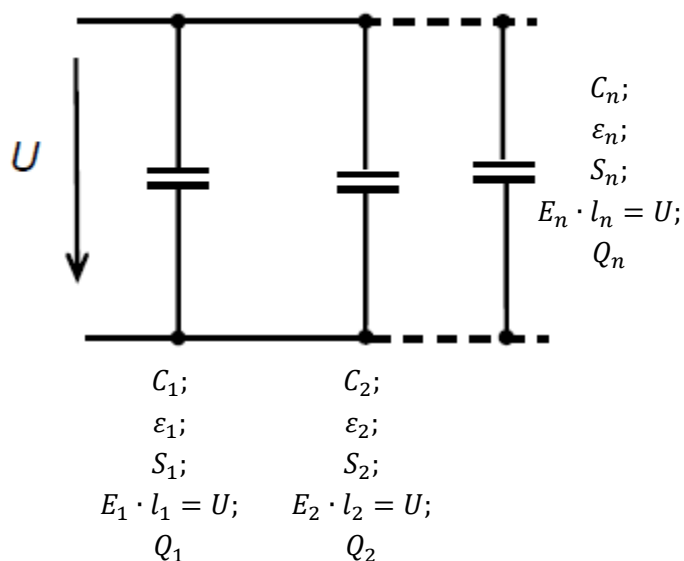
Z druhého KZ je zřejmé, že musí platit

$$U = \sum U_i = Q \cdot \sum \frac{1}{C_i} = \frac{Q}{C_s}$$

Odsud je zřejmé, že k dispozici je pouze jeden náboj  $Q$  (náboje na horní ploše  $C1$  a dolní ploše  $Cn$ ; ostatní náboje jsou pouze indukovány, při vybíjení se kompenzují). Ekvivalentní kapacita sériové struktury je definována vztahem

$$\frac{1}{C_s} = \sum_i \frac{1}{C_i}$$

### Paralelní řazení kapacit (schéma na obr. 13)



Obr. 13 – paralelní řazení kapacit.

Z fyzikální podstaty paralelního řazení je zřejmé, že na všech kondenzátorech je stejné napětí. Pro náboje kondenzátoru proto platí vztah

$$Q_i = C_i \cdot U$$

a celkový náboj paralelní struktury je

$$Q = \sum Q_i = U \cdot \sum C_i = Q \cdot C_p$$

Ekvivalentní kapacita paralelní struktury je definována vztahem

$$C_p = \sum_i C_i$$

## 4.2 Stávající technické provedení laboratorní úlohy

Model kapacitního snímače, který je součástí řešené laboratorní úlohy, je v současné době realizován podkladovou dřevěnou deskou, na níž jsou pevně umístěna rovnoběžně se vzájemnou vzdáleností rovnou šířce měřících elektrod dvě pravítka ve funkci měřidla vzdálenosti a zároveň fyzických vodičků pohyblivé elektrody. Pevná elektroda obdélníkového

tvaru o rozměrech 100 x 200 mm je realizována měděným pásem umístěným na podkladové desce mezi pravítky překrytým průhlednou fólií. Pohyblivá elektroda o maximální šířce shodné s šířkou pevné elektrody je realizována kuprextitem s měděnou plochou o požadovaném tvaru (určuje tvar pohyblivé elektrody a potažmo výslednou změřenou charakteristiku) přilepeném rovněž na dřevěném podkladu.

Měřicí postup pak zahrnuje položení pohyblivé elektrody mezi vodítka do požadované polohy na elektrodu pevnou, zatížení pohyblivé elektrody vhodným závažím tak, aby došlo k co nejtěsnějšímu kontaktu obou ploch, čili ke vzniku kondenzátoru s fóliovým dielektrikem, a následuje samotný akt změření kapacity LCR-mostem HM8118.

#### **4.2.1 Nedostatky stávajícího řešení**

Stávající mechanické řešení modelu kapacitního snímače zanáší do prováděného měření vysokou chybu vzhledem ke snímané kapacitě, která se pohybuje řádově ve stokách pF.

Chyba vzniká především přímo v naměřené hodnotě kapacity a to nekonstantní vzdáleností elektrod modelu kapacitního snímače (dochází ke zvlnění fólie ve funkci dielektrika) a tím zároveň ke změnám struktury dielektrika, které není ideálně pouze fóliové, ale je složeno z fólie a vzduchu v různých poměrech při postupných měřicích pokusech.

Druhou, minoritní, avšak také významnou chybou může být nepřesné odečtení polohy elektrody z instalovaných pravítek.

Oba tyto základní nedostatky by měly být maximálně eliminovány dále navrhovaným řešením.

## 5 Blokový návrh topologie a konstrukčního řešení systému

Řídicí systém úlohy je navržen tak, aby bylo možno provádět měření minimálně ve stejném rozsahu jako před rekonstrukcí úlohy a zároveň s maximální eliminací výše popsaných chyb, které do výsledků měření vnáší stávající konstrukční provedení modelu kapacitního snímače.

### 5.1 Návrh nového konstrukčního provedení úlohy

Úloha bude trvale umístěna v laboratoři VŠB – TUO. Vzhledem k plánovanému vzdálenému řízení bez lidského dohledu na úlohu na místě, je nutné její pevné zakotvení. Úlohu bude možno uchytit buď k desce pracovního stolu, nebo do svislé polohy na zeď. Vzhledem k prostorovým možnostem laboratoře byla zvolena druhá možnost – uchycení kompletní konstrukce úlohy na vhodné podkladové desce na zeď laboratoře.

#### 5.1.1 Návrh nového mechanického provedení kapacitního snímače

Mechanická konstrukce úlohy musí umožňovat změny vzájemné polohy dvou deskových elektrod modelu kapacitního snímače v dostatečně malých krocích tak, že roviny, v nichž se obě elektrody nacházejí musí být rovnoběžné a vzdálenost mezi nimi musí být konstantní (případně nastavitelná). Vzájemná poloha elektrod pak musí být nastavitelná nejméně v takovém rozsahu, aby v krajních polohách nedocházelo k žádnému překrytí obou elektrod modelu kapacitního snímače, resp. docházelo ke vzájemnému překrytí obou elektrod o ploše rovné ploše obou elektrod (nebo menší z elektrod), a to při pohledu kolmém na roviny, v nichž se elektrody nacházejí.

Plocha nových elektrod by měla být zvolena nejméně o velikosti plochy elektrod ve stávajícím konstrukčním řešení úlohy, tedy nejméně  $2 \text{ dm}^2$ .

Pro splnění uvedených požadavků mohou být elektrody koncipovány buď jako půl kruhové (či do tvaru půlprstence), při čemž jedna z elektrod bude pevná, druhá otočná kolem středu kruhu, nebo jako obdélníkové (tvarově shodně se stávajícím stavem) opět s jednou elektrodou pevnou a druhou posuvnou v ose rovnoběžné s delší stranou obdélníku.

Pro vyšší variabilitu využití výsledné konstrukce byla zvolena varianta s posuvnou elektrodou. Pevná elektroda bude ukotvena na podkladovou desku do potřebné výšky, čímž bude definována vzdálenost obou elektrod. Pohyblivá elektroda pak bude upevněna k níže navrženému posuvnému mechanismu. V dráze pohyblivé elektrody, pakliže to délka této dráhy umožní, může být umístěno dokonce více pevných elektrod v různých výškách, čímž je možno demonstrovat vliv změny vzdálenosti obou elektrod kapacitního snímače, dále mohou být použity elektrody o různých tvarech (např. ve tvaru plochy pod exponenciálou apod.).

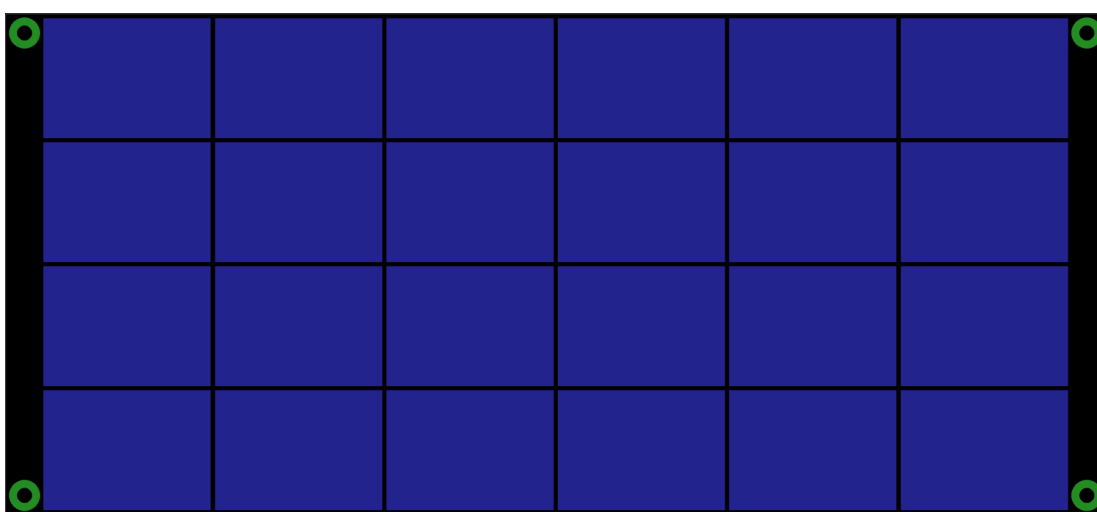
Aby byla variabilita nastavení úlohy ještě vyšší, je možno pohyblivou elektrodu rozdělit na několik obdélníků do podoby matice a umožnit jejich spínání, čímž může být dosaženo velkého množství kombinací tvarů elektrod resp. efektu několika paralelních kondenzátorů.



### 5.1.1.1 Návrh segmentované elektrody a výpočet kapacity

Pohyblivá elektroda bude rozdělena dle obr. 14 na 24 obdélníkových segmentů (záměrně je použitý počet segmentů dělitelný 8 z důvodu efektivního využití posuvných registrů přijímajících signál pro spínání jednotlivých segmentů, více dále v kapitole 6.3), které budou moci být jednotlivě elektricky připínány k měřicímu obvodu a mohou tak prakticky fungovat jako 24 menších kapacitních snímačů.

Celkové rozměry elektrody byly zvoleny tak, aby se blížily rozměrům stávajícího modelu kapacitního snímače. Rozměry jednotlivých segmentů byly zvoleny 25 x 35 mm, celková velikost navržené desky včetně prostoru pro montážní otvory je pak 231 x 105 mm. Mezi jednotlivými segmenty je navržena standardní izolační mezera 1 mm.



Obr. 14 – návrh segmentované elektrody v podobě desky plošných spojů.

Rozměry pevné elektrody modelu kapacitního snímače jsou pak navrženy takové, aby se aktivní vnější okraje obou elektrod kryly, čili 215 x 103 mm.

Maximální kapacita navrženého modelu kapacitního snímače bude tedy dle všech výše uvedených parametrů zjištěna výpočtem 1, kapacita při sepnutí jednoho segmentu pak výpočtem 2:

Z teoretického rozboru vyplývá, že

$$C = \varepsilon_0 \cdot \varepsilon_r \cdot \frac{S}{l}$$

kde

$\varepsilon_0$  je permitivita vakua rovná  $8,85 \cdot 10^{-12} \text{ N} \cdot \text{m}^2 \cdot \text{C}^{-2}$ ,

$\varepsilon_r$  je poměrná permitivita dielektrického prostředí, v tomto případě vzduchu, rovná 1,0006,

$S$  je v tomto případě součet ploch všech 24 segmentů,

$l$  je vzdálenost elektrod modelu kapacitního snímače – uvažujme vzdálenost elektrod 3 mm.

$$C_{max} = 8,85 \cdot 10^{-12} \cdot 1,0006 \cdot \frac{24 \cdot 25 \cdot 10^{-3} \cdot 35 \cdot 10^{-3}}{3 \cdot 10^{-3}}$$

$$\underline{\underline{C_{max} \doteq 62 \text{ pF}}}$$

*Výpočet 1 – celková kapacita navrhovaného modelu kapacitního snímače.*

Kapacita jednoho segmentu elektrody pak může být vypočtena jako

$$C_{segm.} = 8,85 \cdot 10^{-12} \cdot 1,0006 \cdot \frac{25 \cdot 10^{-3} \cdot 35 \cdot 10^{-3}}{3 \cdot 10^{-3}}$$

$$\underline{\underline{C_{segm.} \doteq 2,6 \text{ pF}}}$$

*Výpočet 2 – kapacita jednoho segmentu navrhovaného modelu kapacitního snímače.*

Vzhledem k tomu, že kapacita navrhovaného modelu kapacitního snímače je vzhledem k možné chybě měření poměrně nízká, může být dále po fyzické realizaci modelu vyzkoušeno větší přiblížení obou elektrod, případně vložení dielektrika s vyšší permitivitou mezi elektrody, čímž může být kapacita modelu kapacitního snímače dále zvýšena.

### 5.1.2 Návrh posuvného a pohonného mechanismu

Pohon mechanismu bude řešen motorem. Vzhledem k faktu, že jde o potřebu vykonávat posuvný pohyb, bylo by možno využít lineární motor. Jelikož je však stále celé řešení koncipováno jako nízkonákladové, bude použit motor rotační.

Posuvná elektroda modelu kapacitního snímače bude ukotvena skrze jezdce na dvou rovnoběžných vodících tyčích o kruhovém průřezu a aktivní délce cca 1 m. Další součástí pohonného mechanismu bude závitová tyč spojená s rotorem motoru a s jezdce upevněným k pohyblivé elektrodě modelu kapacitního snímače. Tento mechanismus zajistí převod rotačního pohybu na pohyb posuvný. Bylo by také možné zajistit posuvný pohyb za pomoci ozubeného řemenu umístěného na řemenovém kole motoru, ukotveného k vozíku posuvného mechanismu a napnutého přes druhé řemenové kolo na opačném konci posuvného mechanismu proti řemenovému kolu motoru. Šroubový mechanismus však zajistí vyšší stabilitu, spolehlivost a přesnost posuvu.

Aby bylo možno softwarově znát aktuální polohu posuvné elektrody, bude použit krokový motor v kombinaci s dorazovými spínači. Toto řešení sice vyžaduje po každém spuštění systému automatickou inicializaci polohy vozíku jeho dojetím do jedné z krajních poloh a tím nastavení polohy na známou hodnotu, vzhledem k předpokládané nízké četnosti potřeby takovýchto inicializačních dějů je však řešením nejefektivnějším. V budoucnu je možno konstrukci osadit inkrementálním čidlem polohy či čidlem absolutní polohy. Ve druhém z těchto případů by pak potřeba inicializace krajní polohou byla eliminována.

## 5.2 Blokový návrh řídicí elektroniky

Řídicí, monitorovací a měřicí elektronika nově navrhované úlohy musí obsahovat 5 základních bloků (kompletní hardwarové blokové schéma je na obr. 15):

1. Napájecí zdroj,
2. pohonnou jednotku,
3. zapojení pohyblivé elektrody včetně spínání jejích segmentů,
4. měření kapacity,
5. řídicí jednotku Raspberry Pi.

**Ad 1.** Jak již bylo zmíněno výše, úloha bude umístěna v laboratoři a vzdálený přístup k ní bude žádaný především v době, kdy laboratoř bude bez lidské přítomnosti. Z tohoto důvodu je nutné, aby minimální prostředky pro zajištění síťové komunikace se vzdáleným uživatelem byly napájeny nepřetržitě (výpadky elektrické sítě není nutno řešit). V této konstrukci je jediným a klíčovým prostředkem v místě úlohy nutným pro komunikaci s uživatelem modul Raspberry Pi. První část napájecího zdroje tedy musí zajišťovat stálé napájení o napětí 5 V a možném odebíraném proudu nejméně 0,5 A (vychází ze specifikací USB rozhraní, pomocí něhož je napájení Raspberry Pi obvykle zajišťováno).

Další prostředky napájecího zdroje mohou být sepnuty až ve chvíli, kdy bude uživatelem systému oznámen začátek měření, a naopak po ukončení měření vypnuty.

V době provozu úlohy musí být zajištěno napájení motoru pro posuv o hodnotě cca 3,5 V při zatížení minimálně 1 A, a dále napájení ostatní řídicí logiky 5 V při zatížení do 0,5 A.

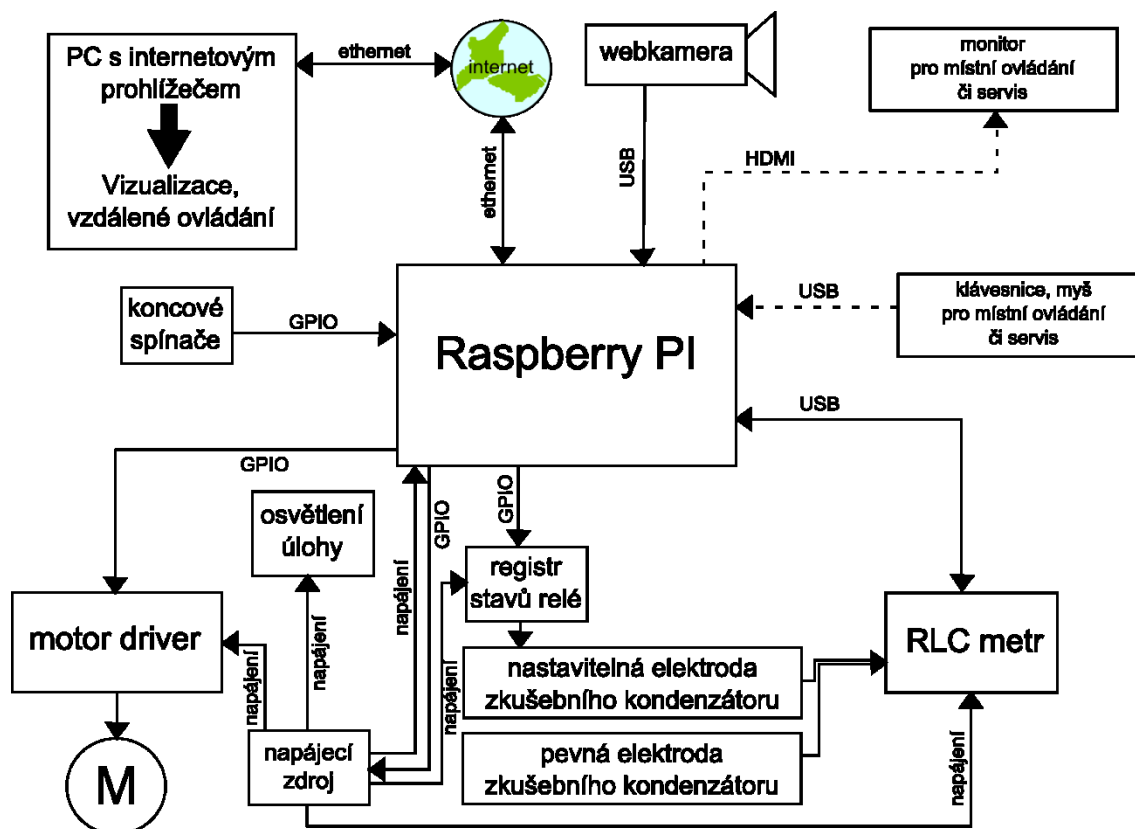
Součástí bloku napájecího zdroje budou také dva triakové síťové spínače zajišťující jednak spínání části napájecího zdroje, která nemusí být pod napětím v době nečinnosti řídicího a měřicího systému, a jednak ke spínání osvětlení úlohy v době, kdy nebude laboratoř dostatečně osvětlena, aby bylo možno získávat uspokojivý obraz činnosti úlohy instalovanou kamerou.

**Ad 2.** Pohonná jednotka posuvného mechanismu musí zajišťovat pootočení závitové tyče mechanické konstrukce o potřebný úhel udaný signálem z Raspberry Pi. Je složena z krokového motoru SPH-54AB-116, výkonových budičů krokového motoru a jejích řídicí logiky.

**Ad 3.** Pohyblivá elektroda modelu kapacitního snímače bude rozdělena na 24 čtvercových či obdélníkových segmentů, které budou moci být libovolně reléově připínány do obvodu modelu kapacitního snímače. Toto bude řešeno samostatnou deskou umístěnou v blízkosti pohyblivé elektrody osazenou 24 relé a řídicí logikou pracující na základě dat odesílaných modulem Raspberry Pi.

**Ad 4.** Měření kapacity bude zajišťováno čtyřvodičovým připojením pevné a pohyblivé elektrody na LCR-most HM8118.

**Ad 5.** Minipočítač Raspberry Pi, jehož technický popis je uveden v kapitole 2.4.1, na němž budou využita rozhraní GPIO (pro spojení s ostatními bloky systému), LAN (pro připojení do sítě internet) a USB (pro připojení webkamery a měřicího přístroje HM8118).

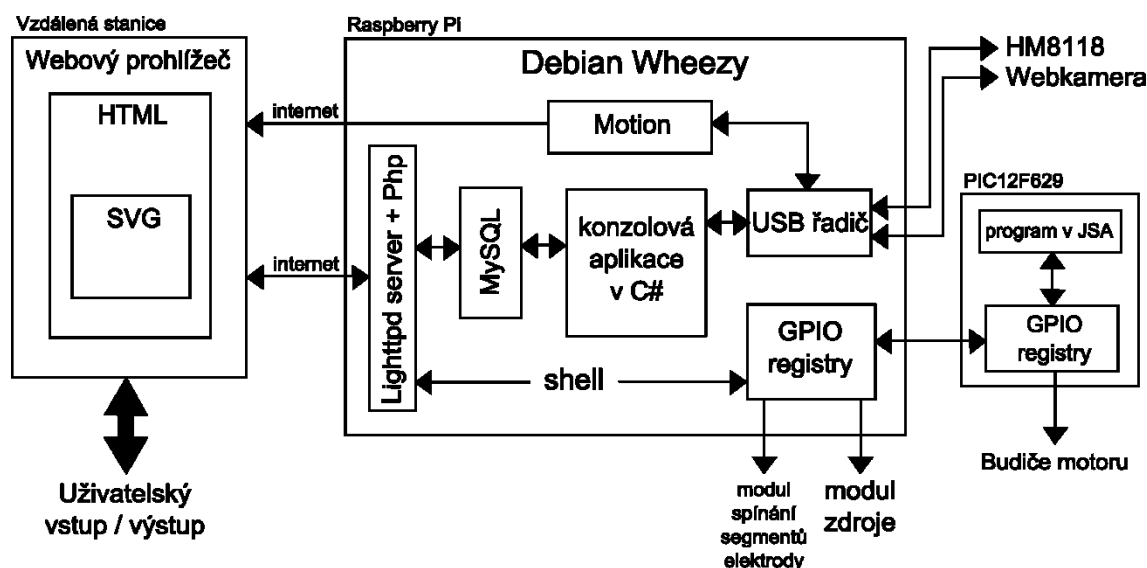


Obr. 15 – hardwarové blokové schéma systému.

### 5.3 Návrh programové topologie systému

Výsledný řídicí a monitorovací systém úlohy měření na modelu kapacitního snímače obsahuje několik zdrojů a výstupů zpracovávaných dat a příkazů, mezi nimiž je nutno zajistit vzájemné transporty informací (kompletní softwarové blokové schéma je na obr. 16):

1. Webová vizualizace a řídicí prvky,
2. výstup webkamery zobrazovaný ve webovém rozhraní,
3. měřicí přístroj HM8118,
4. uložení naměřených dat a dalších aktuálních stavových parametrů systému,
5. Raspberry Pi a jím řízený hardware.



Obr. 16 – softwarové blokové schéma systému.

Webová vizualizace s řídicími a monitorovacími prvky bude jednak zdrojem uživatelských příkazů, které mají za cíl vyvolat mechanickou reakci vozíku pohyblivé elektrody modelu kapacitního snímače, změnu tvaru segmentované elektrody, elektronickou reakci v podobě sepnutí či vypnutí některého elektronického bloku úlohy systému, či samotný akt změření aktuální hodnoty kapacity, případně jiného parametru modelu kapacitního snímače, a jednak výstupem informací o aktuálním mechanickém a elektronickém stavu úlohy, a především výstupem naměřených dat včetně jejich grafického zpracování. Součástí webové aplikace bude také neustále aktuální obraz zaznamenávaný webovou kamerou umístěnou v laboratoři snímající chod úlohy.

Měřicí přístroj HM8118 je vybaven komunikačním rozhraním, skrze které je možné jednak měnit některé parametry měření (měřené veličiny, měřicí rozsahy apod.) a především vyčítat naměřené hodnoty. Naměřené hodnoty je vyžadováno ukládat do níže popsaného datového úložiště, kde budou dostupné k dalšímu zpracování.

Snadno obsluhovatelným datovým úložištěm ve vztahu k webovému rozhraní je databáze založená na SQL, tedy MySQL či PostgreSQL databáze. V tomto případě byla do systému, pod nímž pracuje Raspberry Pi nainstalována databáze MySQL a k ní webové rozhraní pro správu phpMyAdmin. V jednotlivých tabulkách databáze MySQL budou uloženy především naměřené hodnoty, dále může být databáze použita k uložení aktuální polohy pohyblivé elektrody, uložení přihlašovacích informací uživatelů systému, uložení aktuálního stavu sepnutí jednotlivých segmentů pohyblivé elektrody, příp. k uložení dalších informací v případě jakéhokoliv rozšíření systému.

Obrazový výstup webkamery bude odesílán samostatně aplikací Motion na příslušný port serveru běžícího na Raspberry Pi (viz kapitolu 3.1.3).

Komunikace měřicího přístroje HM8118 bude zajištěna fyzicky přes USB rozhraní, programově sériovou komunikací, která bude obsluhována samostatnou konzolovou aplikací běžící na Raspberry Pi. Tato aplikace bude zároveň zajišťovat zápis naměřených hodnot do zmíněné MySQL databáze běžící rovněž na Raspberry Pi.

Příkazy odesílané systému uživatelem pomocí webového rozhraní vyžadující reakci Raspberry Pi změnou hodnoty na GPIO portu mohou být z webové stránky odesílány přímo prostřednictvím PHP, kdy je možno zadat přímo vykonání příkazu Shell interpretu na serveru s operačním systémem Linux (což je zde splněno).

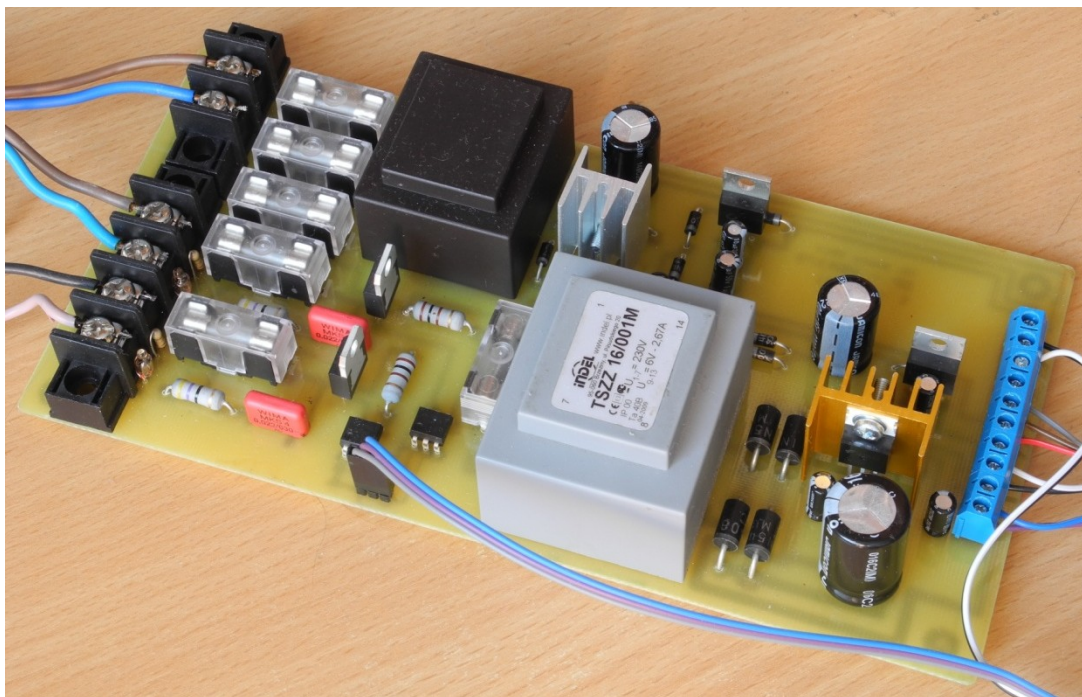
## 6 Funkční popis řídicí elektroniky systému

Jak již bylo zmíněno v blokovém návrhu řídicího systému, řídicí elektronika mimo použité tovární výrobky je složena ze tří základních částí, které zároveň tvoří tři samostatné desky plošných spojů:

1. Napájecí zdroj,
2. pohonná jednotka,
3. řízení elektrického přepínání jednotlivých segmentů pohyblivé elektrody k modelu kapacitního snímače.

### 6.1 Funkční popis bloku napájecího zdroje systému

Deska, na níž je umístěna konstrukce napájecího zdroje (obr. 17) celé úlohy obsahuje kromě nízkonapěťových obvodů zajišťujících potřebná výstupní napětí osazena také triakovými síťovými spínači zajišťujícími spínání osvětlení úlohy, spínání napájení měřicího přístroje HM8118 a současně druhé části napájecího zdroje tak, aby klidový odběr mimo dobu probíhajícího měření byl co možná nejnižší.



Obr. 17 – osazená deska plošných spojů napájecího zdroje.

#### 6.1.1 Napájecí zdroj – nízkonapěťová část

První část napájecího zdroje je konstruována pro nepřetržitý provoz – napájení minipočítače Raspberry Pi, který sám o sobě vyžaduje napájení 5 V při odebíraném proudu až 0,5 A. Po připojení dalších periferních zařízení, v tomto případě webkamery do USB portu,

bude odběr Raspberry Pi navýšen o odběr připojeného zařízení. V tomto případě byl maximální odběr při činnosti webkamery změřen cca 0,6 A. Trvale zapnutá část zdroje je standardní konstrukce s transformátorem, dvojcestným usměrňovačem se 4 diodami v Graetzově zapojení, filtračním kondenzátorem a stabilizátorem s výstupním napětím 5 V zapojeným výrobcem doporučeným způsobem uvedeným v katalogovém listu stabilizátoru. Transformátor je použit v provedení do desky plošných spojů zalitý o standardizovaných rozměrech EI38 s výstupním napětím 6 V o výkonu 4,5 VA. Napětí na nezatíženém usměrňovači s filtračním kondenzátorem je zjištěno výpočtem 3.

$$U_C = \sqrt{2} \cdot (U_{stř.} - 2 \cdot U_{P-N})$$

$$U_C = \sqrt{2} \cdot (6 - 2 \cdot 0,65)$$

$$\underline{\underline{U_C \doteq 6,65 \text{ V}}}$$

*Výpočet 3 – napětí zdroje za usměrňovačem.*

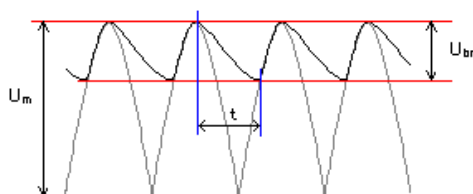
Z vypočtené hodnoty vyplývá, že rozdíl mezi žádaným výstupním napětím zdroje a maximálním napětím na usměrňovači bude nejvýše 1,65 V. Nebude tedy moci být použit běžný stabilizátor řady LM78xx, který má daný minimální rozdíl napětí mezi vstupem a výstupem 2 V. Z dostupných stabilizátorů s nižším minimálním úbytkem napětí byl vybrán typ LM2940CT-5, u nějž je udáván minimální rozdíl napětí na vstupu a napětí na výstupu 0,5 V, při maximálním zatížení 1 A, což je z hlediska naměřeného odběru Raspberry Pi dostačující.

Pro dodržení minimálního úbytku napětí na stabilizátoru je nutné použít takovou hodnotu filtračního kondenzátoru za usměrňovačem, aby napětí na vstupu stabilizátoru v zatíženém stavu na usměrňovači nekleslo oproti napětí ve stavu nezatíženém o více než 1,15 V.

### **Výpočet hodnoty filtračního kondenzátoru [11]**

Pokud bude zvolena nízká kapacita, bude výstupní napětí příliš zvlněné a klesne pod požadovanou minimální hodnotu. Při použití příliš vysoké kapacity může docházet k poruchám vlivem velkého proudového impulsu při zapnutí.

Úkolem filtračního kondenzátoru je akumulovat náboj v době kdy výstupní napětí z usměrňovače dosahuje maxima ( $U_m$ ) a dodávat energii do zařízení v době, kdy výstupní napětí z usměrňovače prochází nulou ( $t$ ). Průběh napětí na kondenzátoru znázorňuje obr. 18.



*Obr. 18 – průběh napětí na filtračním kondenzátoru za usměrňovačem (černou barvou).*



Vlivem střídání doby nabíjení a vybíjení vzniká zvlnění napájecího napětí ( $U_{br}$ ). Napětí na filtračním kondenzátoru při nabíjení kopíruje příslušnou část sinusovky, při vybíjení do ideálního odporu má exponenciální průběh. Nabíjení končí dosažením maximálního napětí (lichá čtvrtina periody), vybíjení končí v okamžiku protnutí obou průběhů:

$$e^{\frac{-t}{R \cdot C}} = \sin(2 \cdot \pi \cdot f \cdot t - 90^\circ)$$

Ve většině případů však zátěž není ideální rezistor a průběh vybíjení není exponenciální. Protože však využíváme jen malou část průběhu vybíjení, můžeme jej zjednodušit jako lineární. Můžeme proto výpočet značně zjednodušit:

$$C = \frac{k \cdot I}{U_{br}}$$

$C$  je kapacita filtračního kondenzátoru [ $\mu F$ ],  
 $I$  je odebíraný proud spotřebiče [ $mA$ ],  
 $U_{br}$  je zvlnění, tedy rozdíl mezi maximálním a minimálním napětím na filtračním kondenzátoru v průběhu periody [ $V$ ],  
 $k$  je konstanta přiřazená maximálnímu dovolenému zvlnění napětí na filtračním kondenzátoru v procentuálním vyjádření ve vztahu k jeho maximální hodnotě v průběhu periody dle tab. 1.

Zvlnění $U_{br}/U_m$ [%]	5	10	15	20	30	40	50
$k$ [-]	9	8,5	8,2	8	7,5	7	6,7

Tab. 1 – přiřazení konstanty  $k$  jednotlivým hodnotám dovoleného zvlnění.

Chyba způsobená zjednodušením výpočtu je menší než výrobní tolerance elektrolytických kondenzátorů.

V tomto případě tedy

$$\frac{U_{br}}{U_m} = \frac{1,15}{6,65} \doteq 17,3 \% \rightarrow k \doteq 8,1$$

$$C = \frac{8,1 \cdot 600}{1,15} \doteq 4226 \mu F$$

Výpočet 4 – potřebná hodnota filtračního kondenzátoru za usměrňovačem.

Zjištěné hodnotě výpočtem 4 bude nejbližší hodnota v řadě 4700  $\mu F$ .

**Druhá část napájecího zdroje** bude napájena ze samostatného transformátoru, který bude v provozu pouze v průběhu měření. Transformátor byl zvolen opět v provedení do desky plošných spojů standardizované velikosti EI54 s výstupním napětím 6 V a maximálním výkonem 16 VA. S ohledem na maximální odběr systému ve stávajícím stavu by bylo možno použít transformátor o výkonu pouhých 10 VA, cenový rozdíl mezi těmito je však pouhých cca 20 %, což vzhledem k možnému budoucímu rozšíření systému o další funkce a vzhledem k ceně transformátoru není problém.

V první řadě bude druhá část zdroje zajišťovat napájení ostatní řídicí logiky rovněž napětím 5 V, kdy proudové zatížení by nemělo překročit hodnotu 0,35 A. Pro jednoduchost a především pro možnost dalšího budoucího rozšíření systému bude pro tento účel použito zapojení shodné s první částí zdroje. Tato část zdroje je dále doplněna zdvojovačem napětí a zdrojem záporného napětí 5 V ve vztahu k zemi celého systému. Zdroj záporného napětí s minimálním proudovým odběrem (řádově jednotky mA) je vyžadován řídicí logikou krokového motoru, tento je koncipován jako standardní zapojení stabilizátoru LM7905 dle katalogového listu.

Druhou úlohou druhé části zdroje je zajišťovat napájení cívek krokového motoru skrze výkonové budiče řídicí elektroniky. Zde je vyžadováno napětí 3,5 V při proudovém odběru 1 A za chodu motoru a cca 0,25 A v případě, kdy je motor v klidu, ale systém je ve stavu probíhajícího měření. Výpočet filtračního kondenzátoru byl proveden analogicky k výpočtu u první části zdroje. Vzhledem k požadavku na nestandardní výstupní napětí zdroje je použit stabilizátor s plynule nastavitelným výstupním napětím pomocí odporového děliče (typ LM317T pro maximální proudové zatížení 1,5 A) dále zapojeného dle doporučení výrobce. V případě, že by byl použit driver motoru, který by používal půlkrokování, tedy napájení dvou sousedních cívek současně při půlkroku, byl by využit plný výkon instalovaného 16VA transformátoru, v usměrňovacím můstku by musely být použity diody s vyšším maximálním proudem (např. typ 1N5408), dále by musela být zvýšena kapacita filtračního kondenzátoru za usměrňovačem přibližně na dvojnásobnou hodnotu a musel by být použit stabilizátor napětí s vyšším maximálním proudovým zatížením (minimálně 2 A, např. typ LM350T).

### 6.1.2 Napájecí zdroj – síťová část

Za účelem minimalizace příkonu celého systému v klidovém stavu jsou vybrané části systému napájeny pouze v případě potřeby.

V prvním případě jde o osvětlení úlohy. Jelikož je požadavek na vizuální monitorování úlohy webovou kamerou, je nutno zajistit také dostatečné osvětlení mechanismu bez ohledu na aktuální hladinu osvětlení v laboratoři, kde bude úloha umístěna. Toto může být zajištěno vhodným svítidlem připojitelným standardně na síť 230 V. Signál k sepnutí či vypnutí osvětlení dá uživatel prostřednictvím ovládacího webového rozhraní. Signál bude interpretován na výstupním pinu Raspberry Pi. Nejjednodušším řešením by bylo spínání pomocí relé, toto by však muselo být doplněno alespoň tranzistorovým spínačem, jelikož maximální proudové zatížení výstupního pinu GPIO portu Raspberry Pi je udáváno 16 mA, což je výrazně nižší hodnota než hodnota potřebná ke spolehlivému sepnutí síťových kontaktů relé. V ideálním případě by pak měl být kontakt s Raspberry Pi s ostatní elektronikou realizován skrze optočlen.

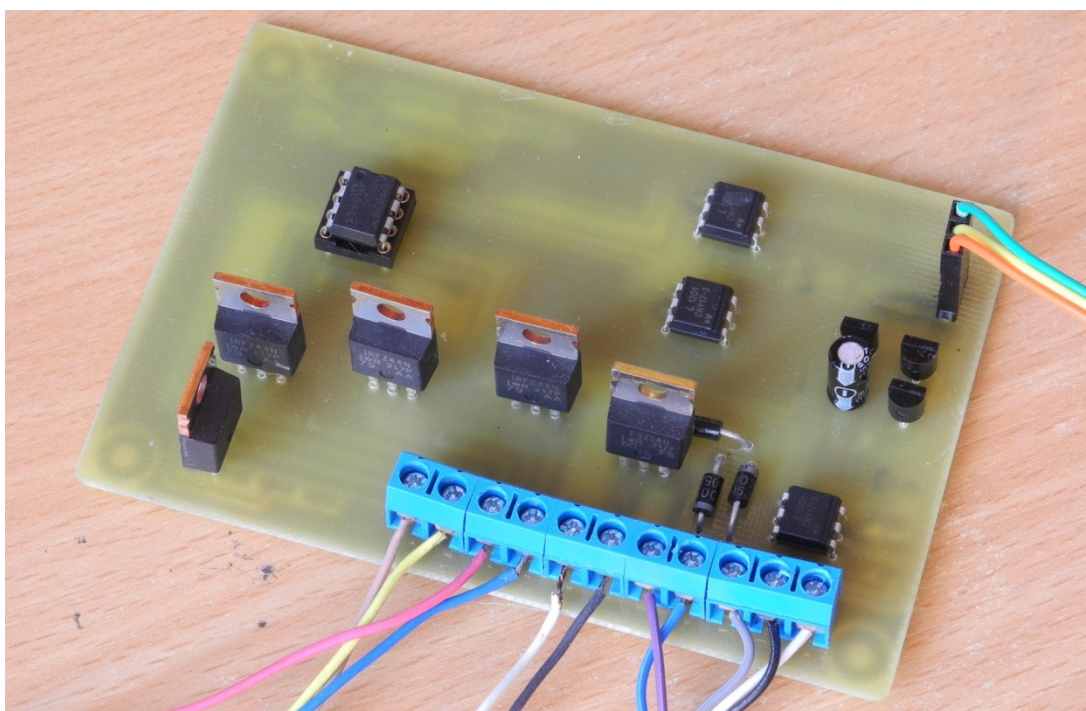
Pokud bude k těmto okolnostem přidán ještě argument omezené elektrické a mechanické životnosti kontaktů relé, je jednoznačně rozhodnuto využít spínání bezkontaktní – triakové.

První částí triakového spínače je optotriak MOC3041 (v pouzdře je integrován obvod spínání při průchodu nulou), který plní funkci jednak galvanického oddělení Raspberry Pi od ostatních obvodů spínače, a jednak funkci signálového spínání síťového napětí. Výstup optotriaku je dále veden do řídicí elektrody výkonového triaku T410-600T, který je již přímou součástí spínaného obvodu osvětlení.

Stejná koncepce je i druhý spínač spínající měřicí přístroj HM8118 a transformátor napájející krokový motor pohonu mechanismu úlohy.

## 6.2 Funkční popis bloku pohonné jednotky mechanismu

Úkolem elektroniky pohonné jednotky osazené na desce dle obr. 19 je zpracování impulsového signálu dodávaného minipočítačem Raspberry Pi skrze jeho GPIO port a na základě tohoto signálu provádět požadované pootočení osy krokového motoru pohánějícího posuvný mechanismus pohyblivé elektrody modelu kapacitního snímače.



*Obr. 19 – osazená deska plošných spojů řídicí elektroniky pohonné jednotky.*

Řídicí signál určený pro pohonnou jednotku přichází ze dvou výstupních GPIO pinů Raspberry Pi, z nichž jeden určuje požadovaný směr pootočení, druhý pak počet kroků pootočení motoru. Oba signály jsou přivedeny na optočleny, které zajišťují jednak galvanické oddělení Raspberry Pi od ostatní elektroniky (ačkoliv toto zde není nezbytné, jde pouze o ochranu GPIO pinů procesoru Raspberry Pi v případě poruchy na straně elektroniky pohonné

jednotky), a jednak zajišťují převod napěťové hladiny Raspberry Pi 3,3 V na napěťovou hladinu 5 V, s níž pracuje zbytek logiky.

Dále jsou vstupní signály vedeny do mikrokontroléru PIC12F629, který je zde použit ve funkci samotného driveru. Funkce programu mikrokontroléru je popsána níže v kapitole 7.1.

Výstupy mikrokontroléru jsou vyvedeny na hradla čtyř výkonových tranzistorů MOSFET IRFZ44, které zajišťují spínání jednotlivých cívek krokového motoru (obr. 20). Tranzistory již ve svém pouzdře obsahují ochrannou diodu proti zpětným proudovým špičkám indukovaným v cívkách motoru při jejich vypínání.

Konstrukce s mikrokontrolérem a samostatnými výkonovými tranzistory byla zvolena z důvodu širších možností přizpůsobení funkčnosti při přibližně stejné výsledné ceně ve srovnání s použitím účelového integrovaného obvodu.

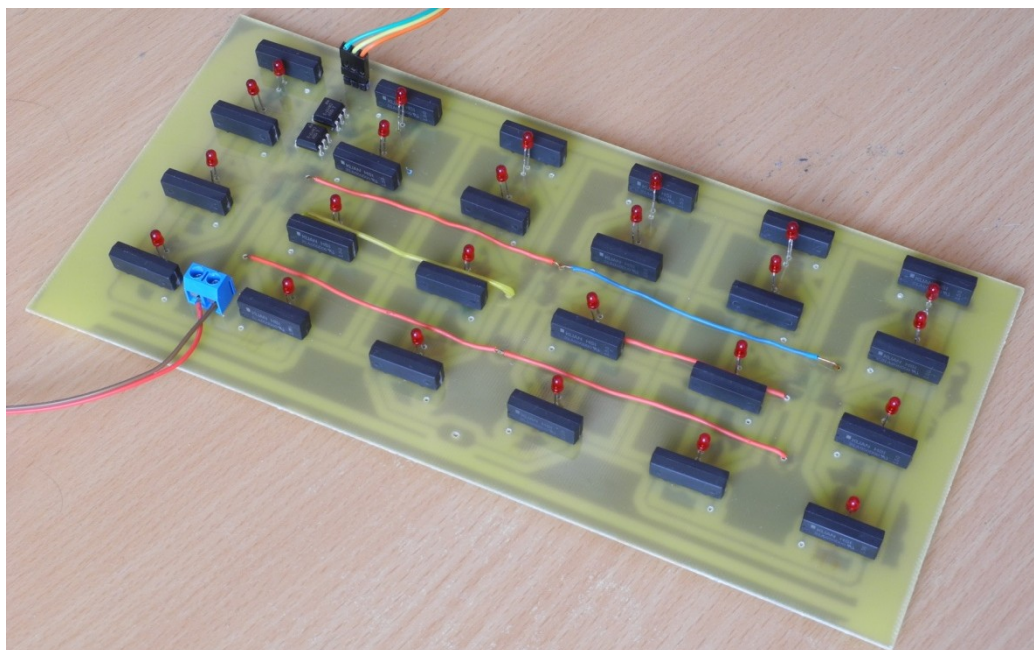
Dále bylo nutno řešit problém napájení krokového motoru v klidovém stavu. Totiž v případě, že by bylo napájení příslušné cívky motoru realizováno pouze za chodu motoru a poté ihned vypnuto, docházelo by po vypnutí napájení k dalšímu nežádoucímu pootočení rotoru vlivem setrvačných sil jak rotoru, tak k němu připojeného šroubového a posuvného mechanismu. Naopak v případě, kdy by byla příslušná cívka krokového motoru napájena stálým plným napětím stejně jako v době, kdy je motor v pohybu, docházelo by po delší době k neúměrnému zahřátí příslušné cívky a v neposlední řadě také ke zbytečně vysokému trvalému odběru proudu ze zdroje. Bylo tedy přistoupeno ke kompromisu, kdy v nehybném stavu je příslušné vinutí udržující motor dostatečnou silou ve stabilní poloze napájeno sníženým napětím, čímž je také snížen jím protékající klidový proud a tudíž také podstatně tepelný ztrátový výkon na motoru. Omezení napětí do vinutí motoru je docíleno vložením sériové kombinace tří diod 1N4007 (pro dosažení potřebného úbytku napětí by bylo možno použít jedinou Zenerovu diodu, tato by však nevyhovovala z důvodu vyššího ztrátového výkonu, který je v realizované variantě rozložen mezi pouzdra tří diod), skrze které je vinutí motoru v klidovém stavu napájeno. Před tím, než se má motor uvést do pohybu, je napětí pro jeho cívky zvýšeno na plnou hodnotu sepnutím tranzistoru IRF9Z34, jímž jsou zmíněné diody překlenuty. Po provedení pohybu je po určitém intervalu (cca 1 s, dáno RC článkem v obvodu pohonné jednotky) tranzistor opět rozepnut a napětí na cívce resp. jí protékající proud se sníží na klidový stav.



*Obr. 20 – použitý krokový motor SPH-54AB-116.*

### **6.3 Funkční popis bloku ovládání segmentované elektrody**

Pohyblivá elektroda modelu kapacitního snímače je složena z 24 segmentů, které mohou být libovolně elektricky připínány k měřicímu obvodu. Spínání jednotlivých segmentů je realizováno pomocí stejného počtu monostabilních mikrorelé. Minimální rozměry relé jsou důležité v první řadě z důvodu minimálního ovlivnění celkové kapacity modelu kapacitního snímače, dále z důvodu minimálního proudového odběru cívky relé a nakonec je také výhodou pro minimální rozměry desky plošných spojů (obr. 21), na níž je tento blok systému realizován.



*Obr. 21 – osazená deska plošných spojů modulu segmentového spínání elektrody.*

Úkolem jednoduché logiky tohoto bloku je zpracování impulsového signálu dodávaného minipočítačem Raspberry Pi skrze jeho GPIO port a na základě tohoto signálu nastavit do sepnutého či rozepnutého stavu jednotlivá relé. Řídící signál přichází ze dvou výstupních GPIO pinů Raspberry Pi – datový a hodinový signál. Oba signály jsou přivedeny na optočleny, které mají stejnou funkci jako v bloku pohonné jednotky. Dále jsou již oba signály vedeny přímo na vstupy třech sériově zapojených osmibitových posuvných registrů 74HC164. Cívky jednotlivých relé pak mohou být připojeny přímo na výstupy posuvných registrů. Proudový odběr cívek totiž při napětí 5 V nepřesahuje 10 mA a v pouzdře relé je dle jeho katalogového listu paralelně k cívce integrována ochranná dioda zabraňující případnému zničení výstupu integrovaného obvodu 74HC164 špičkou zpětného proudu indukovaného v cívce relé při jejím vypínání. Stav každého relé je navíc vhodné indikovat LED diodou, aby však stále nebyl překročen maximální dovolený proud jednotlivých výstupů posuvného registru, byly pro tento účel zvoleny LED s maximálním proudem v propustném směru 2 mA.

## 7 Funkční popis programové části systému

Vyvíjený řídicí a monitorovací systém využívá ke své funkci několik programových prostředků na různých platformách obsluhujících několik hardwarových součástí systému, při čemž některé z prostředků jsou přímo vzájemně provázány, některé naprosto samostatné:

1. Program v mikrokontroléru řídicí elektroniky pohonné jednotky.
2. Webová aplikace pro kontakt s uživatelem a uložení dat.
3. Konzolová aplikace pro komunikaci Raspberry Pi a HM8118.

### 7.1 Funkční popis programu pro mikrokontrolér v modulu pohonné jednotky

Jak již bylo zmíněno v hardwarovém popisu, ke zpracování signálu odesílaného minipočítačem Raspberry Pi do pohonné jednotky je použit osmibitový mikrokontrolér PIC12F629. Signál z Raspberry Pi je do mikrokontroléru přiváděn v podobě dvou bitů, z nichž stav jednoho udává požadovaný směr pootočení motoru, druhý pak v podobě určitého počtu pulsů udává úhel pootočení motoru.

Hlavní program mikrokontroléru je tvořen nekonečnou smyčkou, která pouze neustále kontroluje stav počítadla příchozích impulsů, resp. jeho žádaný stav. V případě, že je z Raspberry Pi odeslán puls na vstup mikrokontroléru, dojde k přerušení a je zavolána funkce pro obsluhu příchozího signálu. Funkce nejprve zjistí požadovaný směr pootočení motoru a na základě toho buď inkrementuje nebo dekrementuje stav počítadla impulsů. Tímto obsluha přerušení skončí a slovo převezme opět hlavní program. V hlavním programu je zkontrolován stav počítadla impulsů (bufferu) a v případě, že je jeho hodnota vyšší či nižší oproti výchozímu stavu, je volána funkce pro pootočení motoru jedním či druhým směrem. Po každém volání funkce pro pootočení motoru je počítadlo o krok vráceno směrem k výchozímu stavu.

Buffer požadovaného počtu kroků pootočení motoru je možné naplnit nejvýše do hodnoty 127 pulsů, což odpovídá pootočení motoru o  $914,4^\circ$ . Jeden základní krok motoru je  $1,8^\circ$ , avšak vzhledem k použitému šroubovému mechanismu, který přesnost posuvu pohyblivé elektrody kapacitního snímače ještě dále zvyšuje, jsou po odeslání jednoho pulsu provedeny hned čtyři kroky motoru (tedy dojde k jednomu proběhu signálu přes všechny čtyři cívky motoru) a dojde tedy k celkovému pootočení osy motoru o  $7,2^\circ$ .

Program pro mikrokontrolér byl vytvořen v JSA<sup>1</sup> pro osmibitové mikrokontroléry PIC.

---

<sup>1</sup> Jazyk symbolických adres (zkratka JSA, anglicky assembly language) nebo také jazyk symbolických instrukcí je v informatice nízkourovňový programovací jazyk, který je tvořen symbolickou reprezentací jednotlivých strojových instrukcí a konstant potřebných pro vytvoření strojového kódu programu pro daný procesor. Symbolickou reprezentaci tvoří zpravidla výrobce procesoru a je založena na mnemotechnických zkratkách, které vyjadřují, co daná strojová instrukce dělá, označují symbolicky registr, slovní zkratku podmínky a podobně. [19]



## 7.2 Funkční popis webové aplikace ve spolupráci s databází jako uložištěm dat

Kompletní webová aplikace využívá ke své činnosti několik softwarových technologií:

1. Značkovací jazyk HTML<sup>2</sup>,
2. skriptovací jazyk PHP<sup>3</sup>,
3. značkovací jazyk SVG<sup>4</sup>,
4. MySQL<sup>5</sup> databázi,
5. linuxový shell<sup>6</sup>.

### 7.2.1 Popis rozložení webového uživatelského rozhraní

Jakmile vzdálený uživatel systému vstoupí do webového uživatelského rozhraní, bude mu zobrazena základní stránka pro ovládání a monitorování systému vytvořená v jazyce HTML a obsahující několik rámců, z nichž v každém je umístěn konkrétní ovládací prvek nebo vizualizace. Rámová struktura stránky je použita z důvodu potřeby automatického obnovování některých částí stránky a naopak nežádoucí je automatické obnovování jiných částí řídicího systému či monitorovacích prvků stránky. Výhodou rámového členění stránky je rovněž absence potřeby znovunačtení celé stránky při obnovování zobrazeného výstupu pouze jednoho monitorovacího prvku resp. jednoho rámu.

Jednotlivé prvky zobrazené na stránce webového uživatelského rozhraní lze rozdělit do tří základních kategorií – čistě monitorovací prvky, čistě ovládací prvky a kombinaci monitorovacích a ovládacích prvků.

První částí uživatelského webového rozhraní je vizualizace části měřicího přístroje – LCR-mostu HM8118 (obr. 22), která je v současné verzi aplikace použita pouze jako monitorovací

---

<sup>2</sup> HyperText Markup Language, označovaný zkratkou HTML, je značkovací jazyk pro hypertext. Je hlavním z jazyků pro vytváření stránek v systému World Wide Web, který umožňuje publikaci dokumentů na Internetu. [20]

<sup>3</sup> PHP (rekurzivní zkratka PHP: Hypertext Preprocessor, česky „PHP: Hypertextový preprocesor“, původně Personal Home Page) je skriptovací programovací jazyk. Je určený především pro programování dynamických internetových stránek a webových aplikací například ve formátu HTML, XHTML či WML. PHP lze použít i k tvorbě konzolových a desktopových aplikací. Pro desktopové použití existuje kompilovaná forma jazyka.

Při použití PHP pro dynamické stránky jsou skripty prováděny na straně serveru – k uživateli je přenášén až výsledek jejich činnosti. [21]

<sup>4</sup> Jazyk SVG (Scalable Vector Graphics) je značkovací jazyk, který vychází z jazyka XML a popisuje vektorovou grafiku. Jedná se o otevřený grafický formát. Výhodou SVG je i to, že se velmi snadno generuje strojově. [8]

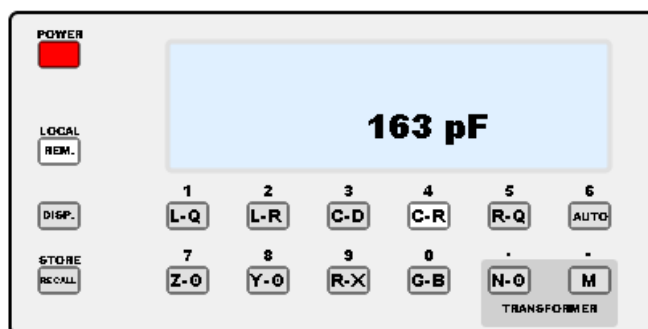
<sup>5</sup> MySQL je multiplatformní databáze. Komunikace s ní probíhá – jak už název napovídá – pomocí jazyka SQL. [22]

SQL je standardizovaný dotazovací jazyk používaný pro práci s daty v relačních databázích. SQL je zkratka anglických slov Structured Query Language (strukturovaný dotazovací jazyk). [23]

<sup>6</sup> Unixový shell (též příkazový procesor, v doslovném překladu „unixová skořápka“) je název textového uživatelského rozhraní, které je předchůdcem grafického uživatelského rozhraní. Shell je spuštěn po přihlášení uživatele do systému, vytvoří příkazový řádek, pomocí kterého uživatel může počítač ovládat a jeho ukončením je uživatel ze systému odhlášen.



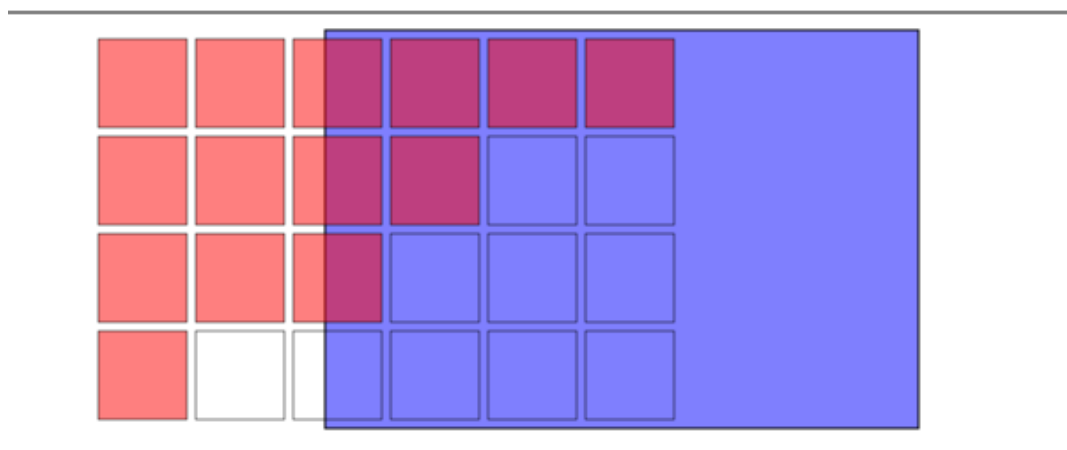
prvek, nicméně bylo by možno vizualizaci jednotlivých ovládacích prvků přístroje přiřadit příslušnou funkci přístroje tak, aby byl přístroj schopen reagovat na uživatelské příkazy a uživatel mohl s přístrojem vzdáleně pracovat takovým způsobem, jako by se nacházel přímo v laboratoři (toto je samozřejmě podmíněno továrními omezenými možnostmi ovládání přístroje při jeho vzdáleném řízení přes komunikační rozhraní).



Obr. 22 – vizualizace základního ovládacího panelu a displeje HM8118.

V další části stránky uživatelského rozhraní je zobrazen obraz snímáný webkamerou umístěnou v laboratoři před monitorovanou úlohou. Bližší popis k vložení obrazu webové kamery do stránky je uveden v kapitole 3.1.3.

Nejrozsáhlejší část webového uživatelského rozhraní tvoří samotné ovládací a vizualizační prvky řízeného a monitorovaného technologického procesu – laboratorní úlohy. Klíčová je vizualizace vzájemného posuvu elektrod modelu kapacitního snímače (obr. 23) spojená s ovládacím rozhraním segmentů pohyblivé (segmentované) elektrody. Vizualizace využívá značkovacího jazyka SVG s výsledkem interaktivní vektorové grafiky.



Obr. 23 – SVG vizualizace pohyblivé elektrody.

Modrá obdélníková plocha symbolizuje první – pevnou – elektrodu modelu kapacitního snímače, červené a bílé plochy sestavené do matice o rozměrech 6 x 4 symbolizují jednotlivé segmenty pohyblivé elektrody, při čemž červeně vyplněné segmenty značí sepnutý stav, bílé segmenty pak stav odpojený od měřicího obvodu. Jednotlivé segmenty lze sepnout resp. vypnout kliknutím na ně ve vizualizaci. Vzájemná poloha obou elektrod modelu kapacitního snímače pak reprezentuje jejich skutečné aktuální vzájemné překrytí.

Další součástí uživatelského rozhraní jsou samotné ovládací prvky (obr. 24).

**Spuštění / vypnutí úlohy:** Přihlásit se, začít měření. Ukončit měření, odhlásit se. ☐ osvětlení úlohy

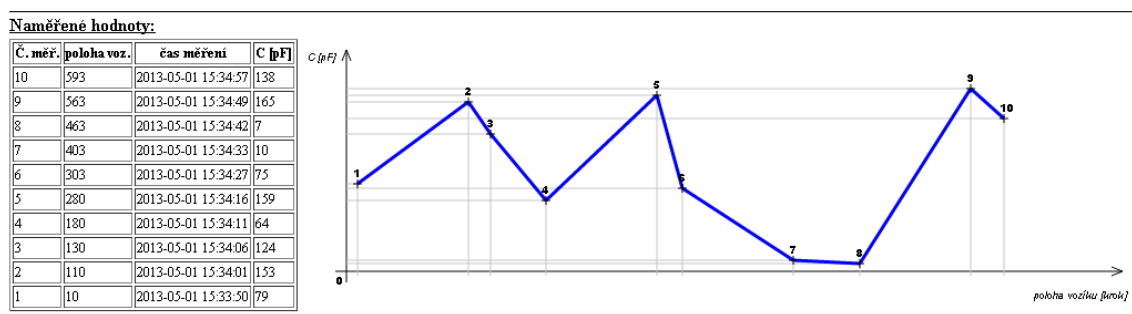
**Posun vozíku:** NAHORU 100 kr. NAHORU 10 kr. NAHORU 1 kr. DOLŮ 1 kr. DOLŮ 10 kr. DOLŮ 100 kr.

**Úkony měření:** ZMĚŘ akt. hodnotu

Obr. 24 – SVG vizualizace pohyblivé elektrody.

Jsou zde celkem 3 řady ovládacích prvků. První řada obsahuje tlačítka pro spuštění resp. ukončení měření a sepnutí osvětlení úlohy. Druhá řada obsahuje pouze ovládací prvky pro posuv vozíku pohyblivé elektrody, tedy posuv oběma směry vždy o potřebný počet kroků. Třetí řada pak obsahuje v současné době pouze jediné tlačítko pro zaznamenání aktuální hodnoty kapacity modelu kapacitního snímače a její zápis do tabulky naměřených hodnot a grafu (viz níže). V budoucnu je možno zde umístit např. také tlačítko pro provedení automatizovaného měření, které provede samočinně stanovený počet měření po stanoveném počtu kroků posuvu pohyblivé elektrody apod.

Poslední částí uživatelského rozhraní ovládání a monitoringu úlohy je zobrazení tabulky naměřených hodnot a k ní náležejícího grafu (obr. 25).

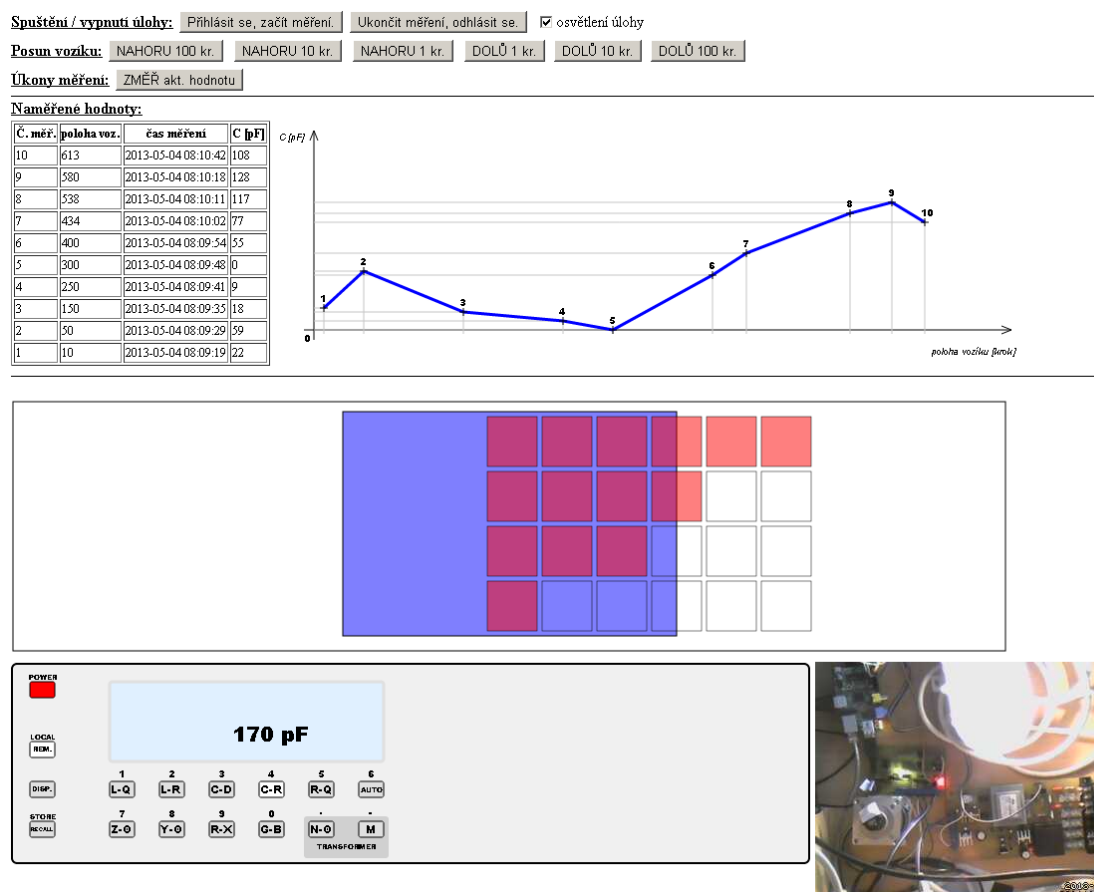


Obr. 25 – tabulka a graf naměřených hodnot.

Tabulka naměřených hodnot obsahuje čtyři údaje o každém měření: Pořadové číslo měření, aktuální polohu vozíku pohyblivé elektrody, čas provedení měření a konečně naměřenou hodnotu aktuální kapacity modelu kapacitního snímače.

Vpravo od tabulky je také samočinně současně s tabulkou aktualizovaný graf všech naměřených hodnot uvedených v tabulce. Graf je opět tvořen vektorovou grafikou ve formátu SVG.

Kompletní pohled na webové uživatelské rozhraní je ukázán na obr. 26.



Obr. 26 – současná výsledná struktura webového uživatelského rozhraní.

## 7.2.2 Popis datových toků ve vztahu k webovému rozhraní

Jako datové uložisko jak přímo pro naměřená data, tak pro pracovní data aplikace je využívána databáze MySQL, v níž jsou data pro webovou aplikaci velice snadno přístupná v požadovaném formátu, a zároveň je možno k ní přistupovat i z jiných platform.

Využívaná databáze obsahuje několik tabulek. Ve vztahu k naměřeným datům jde o tabulky dvě. Do první z nich je naměřená hodnota zapisována, resp. je neustále přepisována v jediném řádku tabulky, při zapnutém měřicím přístroji průběžně níže popsanou konzolovou aplikací běžící na Raspberry Pi spolu s dalšími identifikačními údaji o aktuálním měření. Z této tabulky s průběžnou hodnotou výsledku měření je hodnota vyčítána opět průběžně webovou aplikací za účelem zobrazení aktuální hodnoty ve vizualizaci měřicího přístroje. Druhá tabulka týkající se naměřených hodnot je již plněna pouze na uživatelský příkaz a obsahuje cíleně naměřené hodnoty, které jsou v uživatelském okamžiku zkopírovány z tabulky s průběžnou naměřenou hodnotou. Z této tabulky výsledných naměřených hodnot jsou pak data vyčítána webovou aplikací za účelem zobrazení tabulky a grafu naměřených hodnot.

Další dvě tabulky v databázi jsou pracovním uložištěm aplikace. Je zde tabulka obsahující vždy pouze jedinou hodnotu, a tou je aktuální poloha vozíku pohyblivé elektrody. Tato hodnota je pak webovou aplikací využívána k zobrazení v u jednotlivých naměřených hodnot a rovněž použita ve vizualizaci vzájemné polohy elektrod modelu kapacitního snímače. Dále databáze obsahuje tabulku, v níž jsou zapsány aktuální stavy sepnutí jednotlivých segmentů pohyblivé elektrody modelu kapacitního snímače. Tuto tabulku využívá jednak vizualizace pohyblivé elektrody a jednak PHP skript, který odesílá data skrze výstupní piny Raspberry Pi do posuvných registrů modulu spínání segmentů pohyblivé elektrody.

Dále je navrženo využívání tabulek databáze k ukládání přihlašovacích informací jednotlivých uživatelů systému a zároveň informace i o tom, zda je aktuálně aplikace některým uživatelem využívána.

Odesílání dotazů MySQL databázi a implementace dat vyčtených z databáze do webové stránky resp. do kontextu s parametry značek použitých jazyků HTML a SVG je prováděno prostřednictvím skriptovacího jazyka PHP.

Odesílání příkazů resp. binárních hodnot na GPIO piny Raspberry Pi stejně jako čtení stavů pinů tohoto portu je realizováno pomocí příkazů linuxového shellu, při čemž ve webové aplikaci je toto realizováno opět prostřednictvím skriptovacího jazyka PHP, konkrétně prostřednictvím příkazu `shell_exec()`, který provede na linuxovém serveru příkaz uvedený v jeho parametru.

### **7.3 Funkční popis konzolové aplikace pro Raspberry Pi**

K Raspberry Pi je skrze USB rozhraní připojen měřicí přístroj (LCR-most) HM8118. Komunikace s přístrojem po programové stránce probíhá jako standardní obousměrná sériová komunikace prostřednictvím odesílání resp. přijímání příkazů resp. odpovědí (hodnot) v podobě ASCII znaků. Kompletní přehled dostupných funkcí resp. příkazů je uveden v dokumentaci výrobce k měřicímu přístroji.

Průběh komunikace mezi měřicím přístrojem HM8118 a minipočítačem Raspberry Pi zajišťuje jednoúčelová konzolová aplikace vytvoření v jazyce C# ve vývojovém prostředí MonoDevelop. Měřicí přístroj je v provozu, jak již bylo uvedeno v kapitole 6.1.2, pouze po přihlášení uživatele do ovládacího rozhraní a spuštění měření. Proto i spuštění komunikace s měřicím přístrojem je vhodné limitovat tímto argumentem. Ačkoliv je tedy zde popisovaná konzolová aplikace spouštěna automaticky při spuštění operačního systému, tak tato průběžně kontroluje obsah souboru v systému Raspberry Pi, v němž je definován stav GPIO pinu spínajícího napájení měřicího přístroje. V případě, že je přístroj vypnutý, aplikace žádnou jinou činnost nevykonává. V případě, že je detekováno zapnuté napájení měřicího přístroje, aplikace mu průběžně neustále odesílá dotaz na aktuálně změřenou hodnotu kapacity na měřeném modelu kapacitního snímače. Jakmile měřicí přístroj odešle změřenou hodnotu Raspberry Pi, aplikace ji neprodleně zapíše do příslušné tabulky databáze MySQL spolu s dalšími nezbytnými údaji (aktuální čas, poloha vozíku pohyblivé elektrody modelu kapacitního snímače). Celý tento proces je neustále opakován, dokud není detekováno vypnutí měřicího přístroje nebo dokud není aplikace ukončena.

## 8 Návrh řešení zabezpečení uživatelského přístupu

Uživatelský přístup do monitorovacího a řídicího systému laboratorní úlohy musí být podmíněn autorizací uživatele, který se do systému pokusí vstoupit. Tento přístup může být založen samostatně na principu zadání přístupových údajů (obvykle identifikačního jména a hesla k němu přiřazenému) každého jednotlivého uživatele pro tuto samostatnou aplikaci. Pro uživatele, v tomto případě konkrétně studenta univerzity, by však bylo mnohem efektivnější, kdyby mohl pro přístup k úloze využít přístupových údajů stejných, jako využívá pro přístup do ostatních systémů v rámci svého studia na univerzitě.

Ačkoliv druhé z uvedených řešení by bylo vhodnější, vyžaduje mnohé znalosti mimo rámec této práce a bude řešeno samostatně v závěrečné práci studenta na katedře informatiky.

Dále se tedy návrh bude zabývat pouze prvním navrženým způsobem řešení. V prvé řadě je nutno přiřadit každému uživateli, který má mít umožněn přístup do systému, jeho vlastní uživatelské jméno a přístupové heslo. Tyto údaje musí být pro všechny uživatele systému uchovány ve vhodném uložišti. Údaje je možné ukládat buď v samostatném souboru na serveru, nebo v databázi. Vzhledem k tomu, že zde řešený systém již využívá pro jiné účely databázi MySQL, je vhodné použít tuto. Zároveň musí být heslo zašifrováno tak, aby i přes případné nežádoucí zobrazení obsahu uložistě těchto dat neoprávněným uživatelem, bylo zabráněno vyzrazení přístupových hesel ostatních uživatelů. Za účelem převedení hesla do šifrované podoby může být využit šifrovací algoritmus MD5<sup>7</sup>, kterého je možno dosáhnout použitím stejnojmenné funkce skriptovacího jazyka PHP. V databázi bude tedy vytvořena nová tabulka o minimálně dvou sloupcích – uživatelské jméno a MD5 otisk uživateleova hesla.

Dále je nutno upravit webové uživatelské rozhraní monitorovacího a řídicího systému tak, aby vstupu uživatele do systému byla předřazena stránka pro zadání uživatelských přihlašovacích údajů. Jakmile uživatel zadání provede, údaje budou vyhodnoceny ve smyslu shody kombinace zadaného uživatelského jména a MD5 otisku zadaného hesla s některým řádkem tabulky s údaji uživatelů v MySQL databázi. V případě nalezení identického záznamu v databázi pak může být uživatel přesměrován přímo na řídicí stránku webové aplikace pro práci s laboratorní úlohou, v opačném případě bude vyzván k opětovnému zadání správných přihlašovacích dat.

---

<sup>7</sup> Šifrovací algoritmus MD5 se prosadil do mnoha aplikací (např. pro kontrolu integrity souborů nebo ukládání hesel). MD5 je popsán v internetovém standardu RFC 1321 a vytváří otisk o velikosti 128 bitů. [24]

Po přihlášení uživatele však nestačí pouze přesměrování jej do aplikace, ale jednak musí být uživatelův vstup do systému zaznamenán na straně serveru pro případ, kdy by se do systému pokusil v danou chvíli vstoupit jiný uživatel (nesmí být umožněno současné řízení systému více než jedním uživatelem), což je provedeno opět zápisem příslušné informace do určené tabulky v databázi, a jednak musí být uživatelova příslušnost k systému ověřována při každém načtení stránky s ovládacím rozhraním. Toto může být vyřešeno záznamem do cookies<sup>8</sup> webového prohlížeče tak, že během procesu přihlášení je do počítače uživatele zapsán určitý údaj, který je při každém dalším načtení dané stránky touto stránkou přečten a tím je uživatelova identita ověřena. Aby nedocházelo k trvalému či zbytečně dlouhému zablokování systému jedním uživatelem, je nutno stanovit časový limit doby přihlášení jednoho uživatele (např. 30 min, což je pro měření na úloze dostačující). Odhlášení uživatele proběhne buď na jeho vlastní příkaz, nebo po uběhnutí nastaveného časového limitu a to vypršením doby platnosti cookies v počítači uživatele a zároveň vypršení platnosti záznamu v databázi o uživatelově přihlášení.

V případě, že je již v systému přihlášen jeden uživatel a dojde k pokusu o přihlášení uživatele dalšího, může mu být umožněn vstup do systému, avšak do pozice pozorovatele s přístupem pouze k monitorovacím funkcím. O tomto „read only“ režimu musí být uživatel informován přímo ve webovém rozhraní, případně také ještě před přihlášením do systému.

Z časových důvodů byl tento systém zabezpečení uživatelského přístupu pouze navržen, nikoliv realizován. Jak již však bylo vyhodnoceno na počátku této kapitoly, efektivnější přístupový systém by měl být vyvinut na katedře informatiky.

---

<sup>8</sup> Jako cookie se v protokolu HTTP označuje malé množství dat, která WWW server pošle prohlížeči, který je uloží na počítači uživatele. Při každé další návštěvě téhož serveru pak prohlížeč tato data posílá zpět serveru. Cookies běžně slouží k rozlišování jednotlivých uživatelů, ukládají se do nich uživatelské předvolby apod. [25]

## 9 Dosažené výsledky

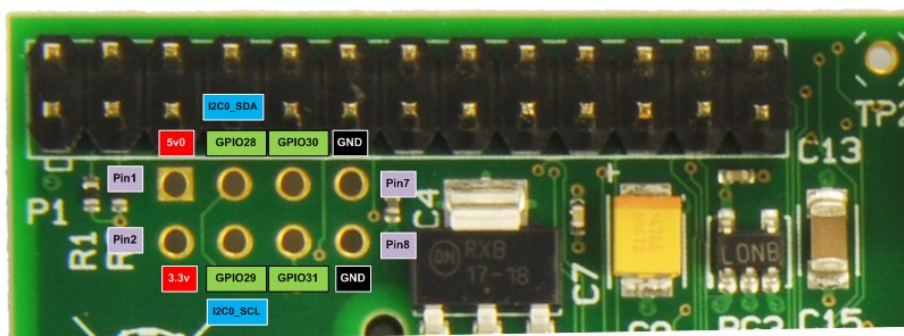
Bylo vytvořeno programové vybavení a zkonstruována kompletní elektronika pro zařízení, pomocí něhož je možné provádět měření na laboratorní úloze „Kapacitní snímače“ studenty předmětu „Senzory a měření“ z libovolného zařízení s připojením k internetu a podporovaným webovým prohlížečem.

Po připojení k webovému ovládacímu rozhraní systému je možné dálkově sepnout napájení elektroniky, prostřednictvím instalované webové kamery sledovat provoz systému, odesílat příkazy k mechanickým úkonům, což je v tomto případě pohyb krokového motoru, a rovněž k elektronickým úkonům, což může být jak již zmíněné sepnutí napájení úlohy, tak elektronické řízení plochy jedné z elektrod modelu kapacitního snímače. Dále je možné sledovat aktuální naměřenou hodnotu instalovaným měřicím přístrojem (RCL mostem) a naměřené hodnoty v požadovaných okamžicích zaznamenávat do tabulky a znázorňovat v grafu implementovaném a automaticky aktualizovaném na webové stránce.

### 9.1 Potenciál dalšího rozvoje systému

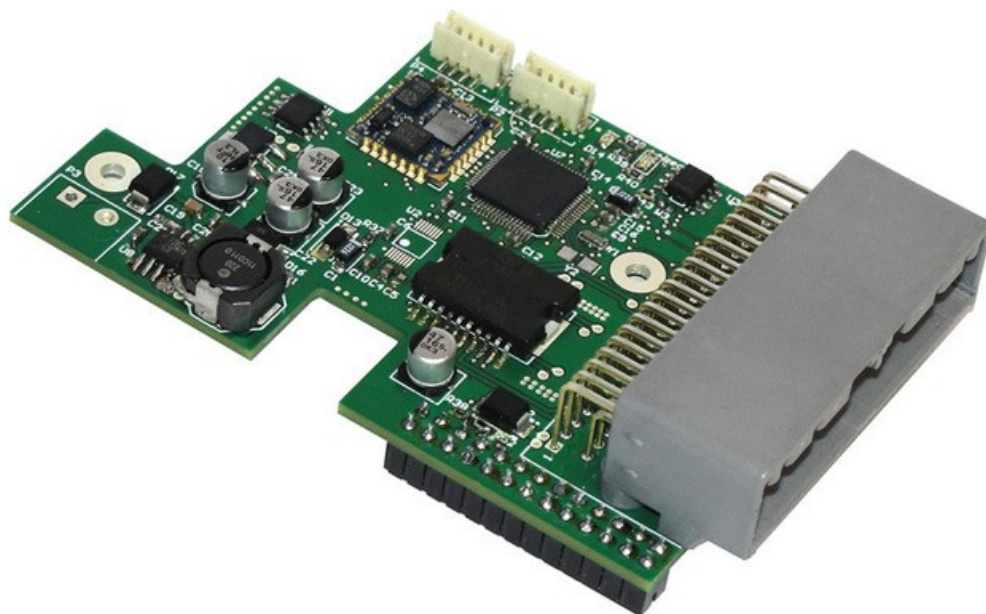
Celé zařízení je umístěno v laboratoři spolu s dalšími úlohami z oboru měření neelektrických veličin. Dalšího přínosu zde bylo možno dosáhnout jeho využitím také ve zmíněných dalších úlohách. Využít by bylo možno jak navrženou mechanickou konstrukci s posuvným systémem, tak funkci samotného vzdáleného monitorování a řízení.

Co se týče výpočetního výkonu použitého minipočítače Raspberry Pi, ten je v současném stavu využit cca z 60 %. Z hardwarového hlediska jsou sice v současném stavu na Raspberry Pi obsazeny všechny výrobně osazené GPIO piny, nicméně Raspberry Pi skýtá další rozšiřující možnosti ať už pouhým fyzickým doplněním pinů do desky plošných spojů dle obr. 27, nebo použitím rozšiřující karty detailně popsané na adrese odkazu [10] a zobrazené na obr. 28. Ve stručnosti rozšiřující karta nabízí další digitální vstupy a výstupy, A/D převodník, drivery možnosti komunikace po dalších rozhraních (RS232, RS485 apod.).



Obr. 27 – možnost rozšíření Raspberry Pi o další GPIO piny. [12]





*Obr. 28 – rozšiřující karta pro Raspberry Pi. [9]*

Ve zmíněné laboratoři jsou měřeny např. úlohy pracující se snímači polohy či s akcelerometrem, které by bylo možno implementovat do vyvinutého systému. Pro úlohu se snímači polohy by mohlo být využito lineárního posuvného mechanismu pro umístění optických snímačů lineárního posuvu, rovněž by bylo možno úlohu doplnit o inkrementální snímač rotačního pohybu, který probíhá v ose krokového motoru mechanismu. Výše zmíněná rozšiřující karta pro Raspberry Pi rovněž volitelně obsahuje akcelerometr, na němž je možno demonstrovat měření zrychlení při posuvu vozíku mechanického systému.

Realizace veškerých těchto dalších aplikací by vyžadovalo další rozšíření hardwaru i softwaru systému, bylo by však možné, efektivní a velice zajímavé např. jako téma další závěrečné práce.

## 10 Závěr

Cílem práce bylo navrhnout, případně zkonstruovat systém, pomocí kterého bude možno vzdáleně monitorovat a řídit zvolený technologický proces, a to s takovými parametry, aby byly možnosti vzdáleného monitorování a řízení maximálně přiblíženy možnostem, které poskytuje lokální řízení a osobní dohled na daný technologický proces.

Na počátku byly definovány požadavky na funkci celého zařízení, poté bylo zváženo několik možností hardwarového provedení a vybráno nejefektivnější z nich v podobě minipočítače Raspberry Pi jako srdce celého systému. Následně byl vyvíjen základní princip systému komunikace mezi serverem a vzdálenou stanicí. Ten je ve výsledku založen na webovém serveru běžícím na Raspberry Pi. Dále byla zprovozněna základní část monitorovacího systému v podobě přenášení obrazu z webkamery na vzdálenou stanicí za pomoci aplikace Motion. Jakmile byl dořešen základní princip komunikace mezi serverem a uživatelskou stanicí, byl vybrán technologický proces vhodný k řízení a monitorování vyvíjeným přípravkem. Byla zvolena laboratorní úloha měření na modelu kapacitního snímače. K úloze bylo navrženo nové mechanické řešení na místo stávajícího nevyhovujícího, následně byla navržena a realizována kompletní nová řídicí elektronika úlohy a po jejím oživení bylo pokračováno ve vývoji programové části systému. Bylo vyvinuto webové uživatelské rozhraní pro řízení a monitorování provozu úlohy včetně návrhu zabezpečení uživatelského přístupu do systému. Současně byla také vyvinuta konzolová aplikace pro Raspberry Pi, která zajišťuje komunikaci systému s připojeným měřicím přístrojem HM8118. Nakonec byly funkce veškerých realizovaných bloků odzkoušeny po hardwarové i softwarové stránce, při čemž nebyl shledán žádný provozní problém.

Výsledkem celého návrhu je možnost provádět měření na laboratorní úloze vzdáleně z jakéhokoliv zařízení s připojením k internetu a webovým prohlížečem podporujícím použité technologie s výsledky v porovnání se stávajícím stavem rovnocennými nebo kvalitnějšími.

Tato práce pro mne byla nejen dlouho očekávaným důležitým krokem k úspěšnému závěru studia, rekapitulací mnoha znalostí během studia nabytých a jejich uvedením do praxe, ale mnohdy velmi zajímavou formou bližšího seznámení se opět s dalšími v mém obzoru novými zajímavými technologiemi, což považuji za pouhé, avšak významné pokračování nekonečného procesu průběžného doplňování vlastních znalostí během profesního i nadšeneckého života každého odborníka ve všech oborech lidské činnosti.

## 11 Použitá a nastudovaná literatura

- [1] BRTNÍK, Bohumil. *Elektrická měření pro bakaláře*. 1. vyd. Praha: BEN - technická literatura, 2011, 170 s. ISBN 978-80-7300-405-7.
- [2] *Elektrotechnická měření*. 1. vyd. Praha: BEN - technická literatura, 2002, 255 s. ISBN 80-730-0022-9.
- [3] HRBÁČEK, Jiří. *Moderní učebnice programování mikrokontrolérů PIC 1. díl*. 1. vydání. [s.l.] : BEN – technická literatura, 2004. 96 s. ISBN 80-7300-136-5.
- [4] LEENHOUTS, Albert C. *Step motor system design handbook*. second edition. United States of America: Litchfield Engineering Co., Kingman AZ U.S.A., 1997. ISBN 1-881855-08-2.
- [5] PUNČOCHÁŘ, Josef a Jitka MOHYLOVÁ. *Elektrické obvody I*. Ostrava, 2007. Studijní materiály pro studijní obor Biomedicínská technika. VŠB – Technická univerzita Ostrava.
- [6] *Geeky Gadgets* [online]. [cit. 2013-05-07]. Sweetbox. URL: <<http://www.geeky-gadgets.com/wp-content/uploads/2013/01/Sweetbox-Raspberry-Pi-Case.jpg>>
- [7] *Hw.cz* [online]. [cit. 2013-05-07]. Vývojový kit. URL: <<http://www.hw.cz/files/autor3994/files/PC010011.jpg>>
- [8] Ing. Vojtěch HORDEJČUK. Hordějčuk, Vojtěch. Jazyk SVG [online]. [cit. 2013-05-07]. URL: <<http://voho.cz/wiki/informatika/jazyk/svg/>>
- [9] *Kick starter* [online]. [cit. 2013-05-07]. Raspberry Pi extend card. URL: <<https://s3.amazonaws.com/ksr/projects/507426/photo-main.jpg?1366748031>>
- [10] *Kick starter* [online]. [cit. 2013-05-07]. Smart IO Expansion Card for Raspberry PI. URL: <<http://www.kickstarter.com/projects/95547492/smart-io-expansion-card-for-raspberry-pi>>
- [11] *Kutilův koutek* [online]. [cit. 2013-05-07]. Filtrační kondenzátor. URL: <<http://www.dmaster.wz.cz/postupy/filtr/filtr.htm>>
- [12] *Mark's space dot com* [online]. [cit. 2013-05-07]. Expand Raspberry Pi GPIO pins. URL: <<http://marksspacedotcom.files.wordpress.com/2013/04/gpio-p51.png>>
- [13] *MOOVIKA* [online]. [cit. 2013-05-07]. Aluminum box. URL: <<http://www.moovika.fr/raspberry-pi.fr/wp-content/uploads/DSC01812-P-shop-436x270.jpg>>
- [14] *National Instruments Česká republika a Slovensko* [online]. [cit. 2013-05-07]. Měřicí karta. URL <[http://sine.ni.com/images/products/us/040806\\_pci6251\\_1.jpg](http://sine.ni.com/images/products/us/040806_pci6251_1.jpg)>

- [15] *Pandatron.cz : Elektrotechnický magazín* [online]. c2000 - 2011 [cit. 2011-04-29]. Škola programování PIC-1. - 13. díl. URL: <<http://pandatron.cz/?elektronika&mi&page=15>>. ISSN 1803-6007.
- [16] *Retromaster's Electronics Projects* [online]. [cit. 2013-05-07]. PIC block diagram. URL: <<http://retromaster.files.wordpress.com/2010/05/mipsPIC32blockdiagram.jpg>>
- [17] *Tecomat* [online]. [cit. 2013-05-07]. Tecomat foxtrot programmable logic controllers: <[img.docstoccdn.com/thumb/orig/20402260.png](http://img.docstoccdn.com/thumb/orig/20402260.png)>
- [18] *Wikipédia* [online]. 2011-02-06 [cit. 2013-05-07]. MOSFET. URL: <<http://sk.wikipedia.org/wiki/MOSFET>>.
- [19] *Wikipédia* [online]. 2013-05-06 [cit. 2013-05-07]. Jazyk symbolických adres. URL: <[http://cs.wikipedia.org/wiki/Jazyk\\_symbolick%C3%BDch\\_adres](http://cs.wikipedia.org/wiki/Jazyk_symbolick%C3%BDch_adres)>
- [20] *Wikipédia* [online]. 2013-03-20 [cit. 2013-05-07]. HyperText Markup Language. URL: <[http://cs.wikipedia.org/wiki/HyperText\\_Markup\\_Language](http://cs.wikipedia.org/wiki/HyperText_Markup_Language)>
- [21] *Wikipédia* [online]. 2013-04-30 [cit. 2013-05-07]. PHP. URL: <<http://cs.wikipedia.org/wiki/Php>>
- [22] *Wikipédia* [online]. 2013-03-07 [cit. 2013-05-07]. MySQL. URL: <<http://cs.wikipedia.org/wiki/Mysql>>
- [23] *Wikipédia* [online]. 2013-03-07 [cit. 2013-05-07]. SQL. URL: <<http://cs.wikipedia.org/wiki/SQL>>
- [24] *Wikipédia* [online]. 2013-03-09 [cit. 2013-05-07]. Message-Digest algorithm. URL: <<http://cs.wikipedia.org/wiki/MD5>>
- [25] *Wikipédia* [online]. 2013-04-15 [cit. 2013-05-07]. HTTP cookie. URL: <<http://cs.wikipedia.org/wiki/Cookies>>
- [26] *ZDNet* [online]. [cit. 2013-05-07]. Raspberry Pi. URL: <<http://cdn-static.zdnet.com/i/story/70/00/005919/raspberry-pi-b-256.jpg>>

## 12 Seznam příloh

Příloha A: Seznam použitých součástek pro výrobu elektroniky.

Příloha B: Schémata zapojení:

- 1 Napájecí zdroj;
- 2 řízení motoru;
- 3 spínání segmentované elektrody.

Příloha C: Předlohy desek plošných spojů.

- 1 Napájecí zdroj;
- 2 řízení motoru;
- 3 spínání segmentované elektrody;
- 4 segmentovaná elektroda;
- 5 pevná elektroda.

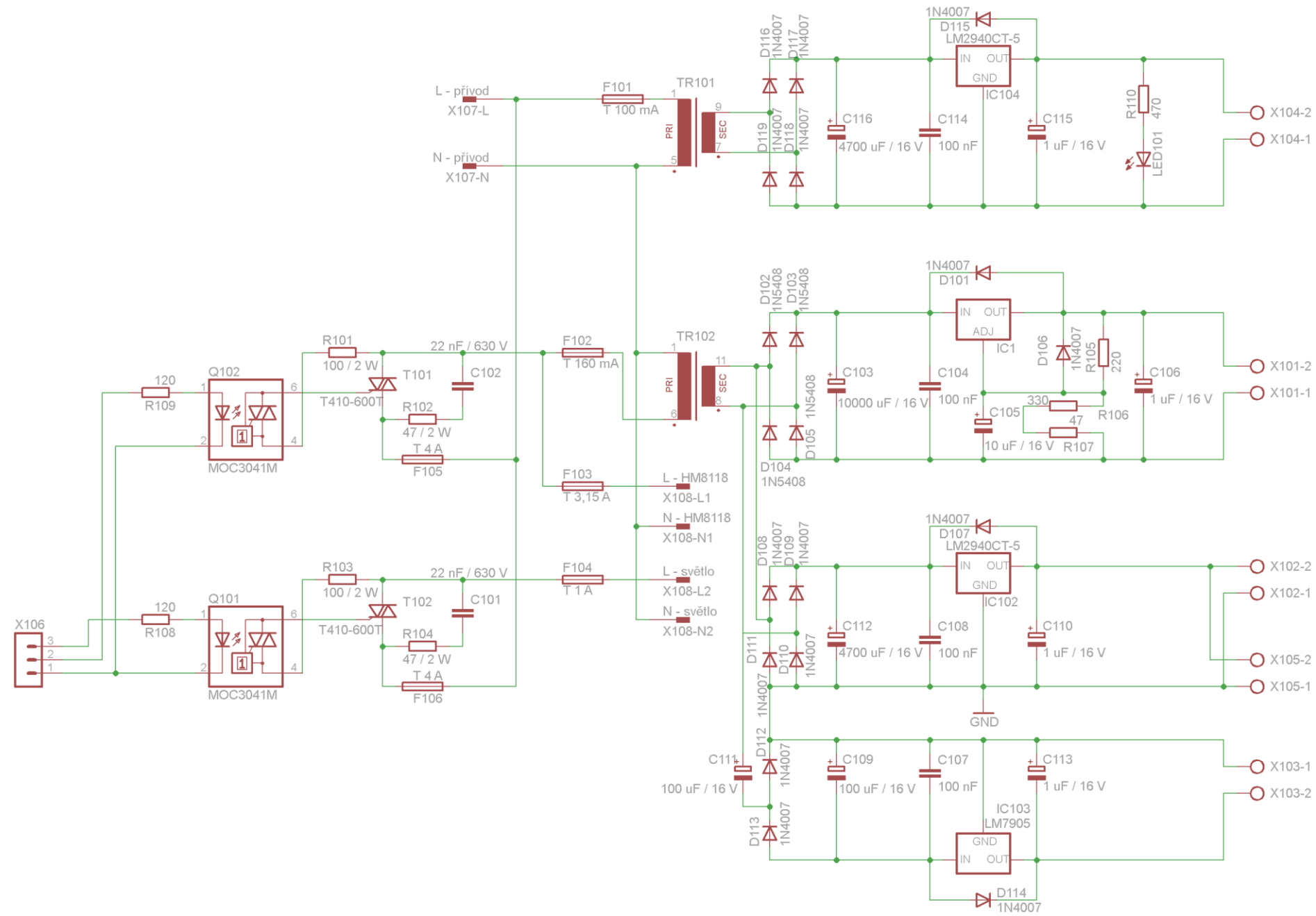
Příloha D: Obsah DVD s touto prací.

## Příloha A

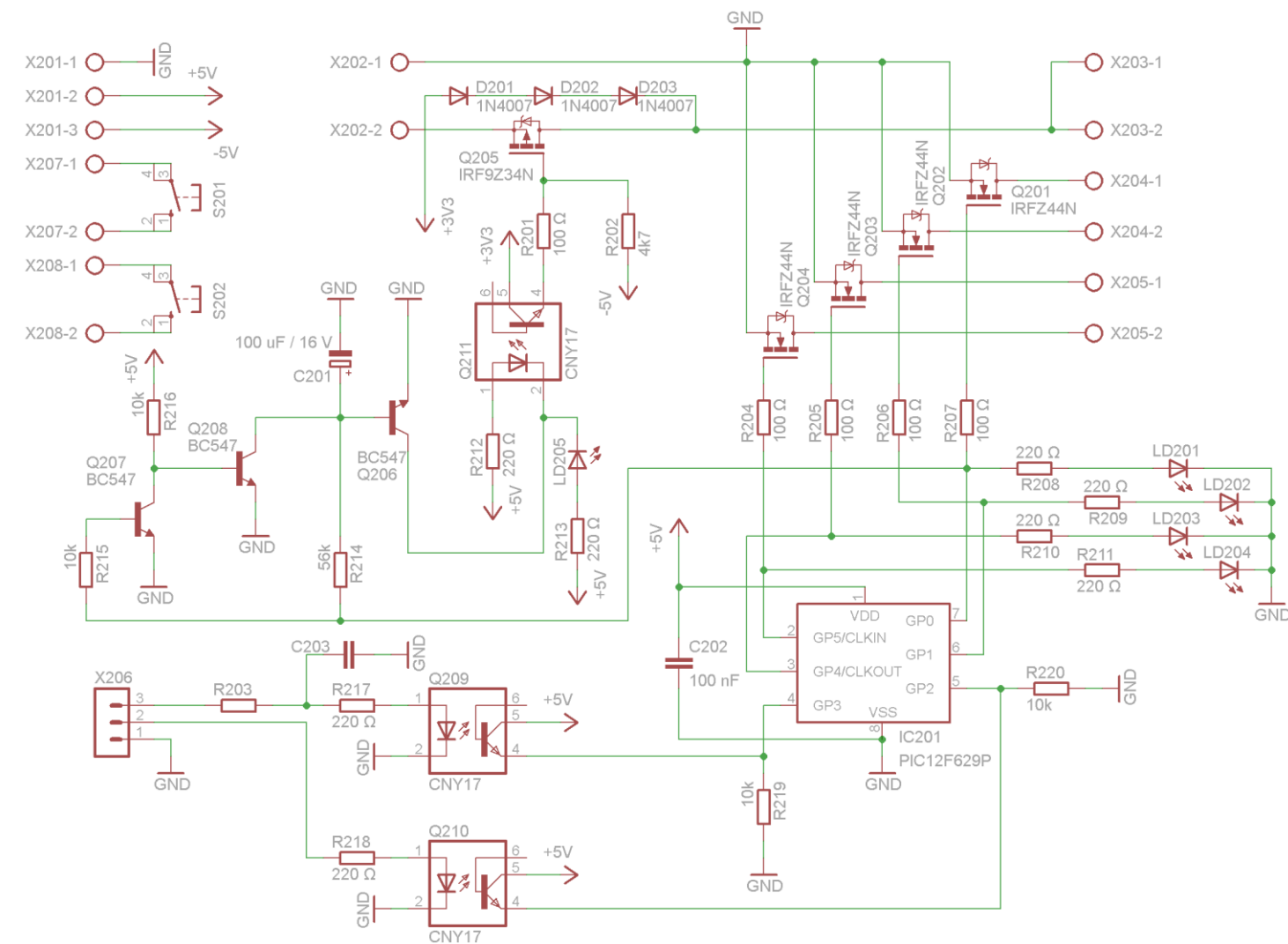
Název součástky	hodnota	počet kusů	cena za všechny kusy
Rezistor	100 Ω / 2 W	2	6,00 Kč
Rezistor	47 Ω / 2 W	2	6,00 Kč
Rezistor SMD 1206	100 Ω	5	4,55 Kč
Rezistor SMD 1206	220 Ω	11	10,01 Kč
Rezistor SMD 1207	330 Ω	1	0,91 Kč
Rezistor SMD 1209	10 kΩ	6	5,46 Kč
Rezistor SMD 1208	47 Ω	1	0,91 Kč
Rezistor SMD 1209	120 Ω	2	1,82 Kč
Rezistor SMD 1210	470 Ω	1	0,91 Kč
Rezistor SMD 1206	1,8 kΩ	24	21,84 Kč
Rezistor SMD 1206	4,7 kΩ	1	0,91 Kč
Rezistor SMD 1206	56 kΩ	1	0,91 Kč
Keramický kondenzátor	100 nF	8	7,28 Kč
Fóliový kondenzátor MKS4	22 nF / 630 V	2	10,00 Kč
Elektrolytický kondenzátor	1 μF / 16V	4	4,00 Kč
Elektrolytický kondenzátor	10 μF / 16V	1	1,00 Kč
Elektrolytický kondenzátor	100 μF / 16V	3	3,00 Kč
Elektrolytický kondenzátor	4700 μF / 16V	2	22,80 Kč
Elektrolytický kondenzátor	10 mF / 16 V	1	38,20 Kč
Dioda	1N4007	18	18,00 Kč
Dioda	1N5408	4	6,40 Kč
Trubičková pojistka 5 x 20 mm	T 100 mA	1	3,30 Kč
Trubičková pojistka 5 x 20 mm	T 160 mA	1	3,30 Kč
Trubičková pojistka 5 x 20 mm	T 1 A	1	3,30 Kč
Trubičková pojistka 5 x 20 mm	T 3,15 A	1	3,30 Kč
Trubičková pojistka 5 x 20 mm	T 4 A	2	8,20 Kč
Tranzistor	BC547C	3	3,30 Kč
Tranzistor	IRFZ44N	4	62,00 Kč
Tranzistor	IRF9Z34N	1	16,00 Kč
Relé	S1A050D00	24	506,40 Kč
Stabilizátor napětí	LM317T	1	7,30 Kč
Stabilizátor napětí	LM2940CT-5	2	51,60 Kč
Stabilizátor napětí	LM7905	1	8,30 Kč
Posuvný registr SMD	74HC164	3	22,00 Kč
Mikrokontrolér	PIC12F629P	1	32,10 Kč
Optočlen	CNY17-3	5	22,00 Kč
Optotriak	MOC3041M	2	21,20 Kč
Triak	T410-600T	2	33,00 Kč
LED SMD 1206	20 mA	5	15,00 Kč
LED 3 mm	2 mA	24	82,50 Kč
Zalitý transformátor EI38	230 V / 6 V 4,5 VA	1	98,00 Kč
Zalitý transformátor EI48	230 V / 6 V 16 VA	1	215,00 Kč
Koncový mikrospínač	-	2	10,00 Kč
Svorka ARK500/2	-	10	36,00 Kč
Svorka ARK500/3	-	1	5,50 Kč
Svorka ARK800/2EX	-	1	8,70 Kč
Svorka ARK800/4EX	-	1	17,00 Kč
Konektor - lámací lišta	S1G20	1	4,40 Kč
Dutinka	KONPC-SPK-PI	20	10,00 Kč
Konektor - dutinka	KONPC-SPK-1	11	5,50 Kč
Konektor - dutinka	KONPC-SPK-3	3	3,30 Kč
Chladič	V7141	3	25,80 Kč
Drobný spojovací materiál	-	-	30,00 Kč
Propojovací vodiče	FBK10H	1 m	22,20 Kč
Celková cena:			<b>1 566,41 Kč</b>

Ceny jsou uvedeny dle ceníku GM electronic platného k 1. květnu 2013.

Příloha B1

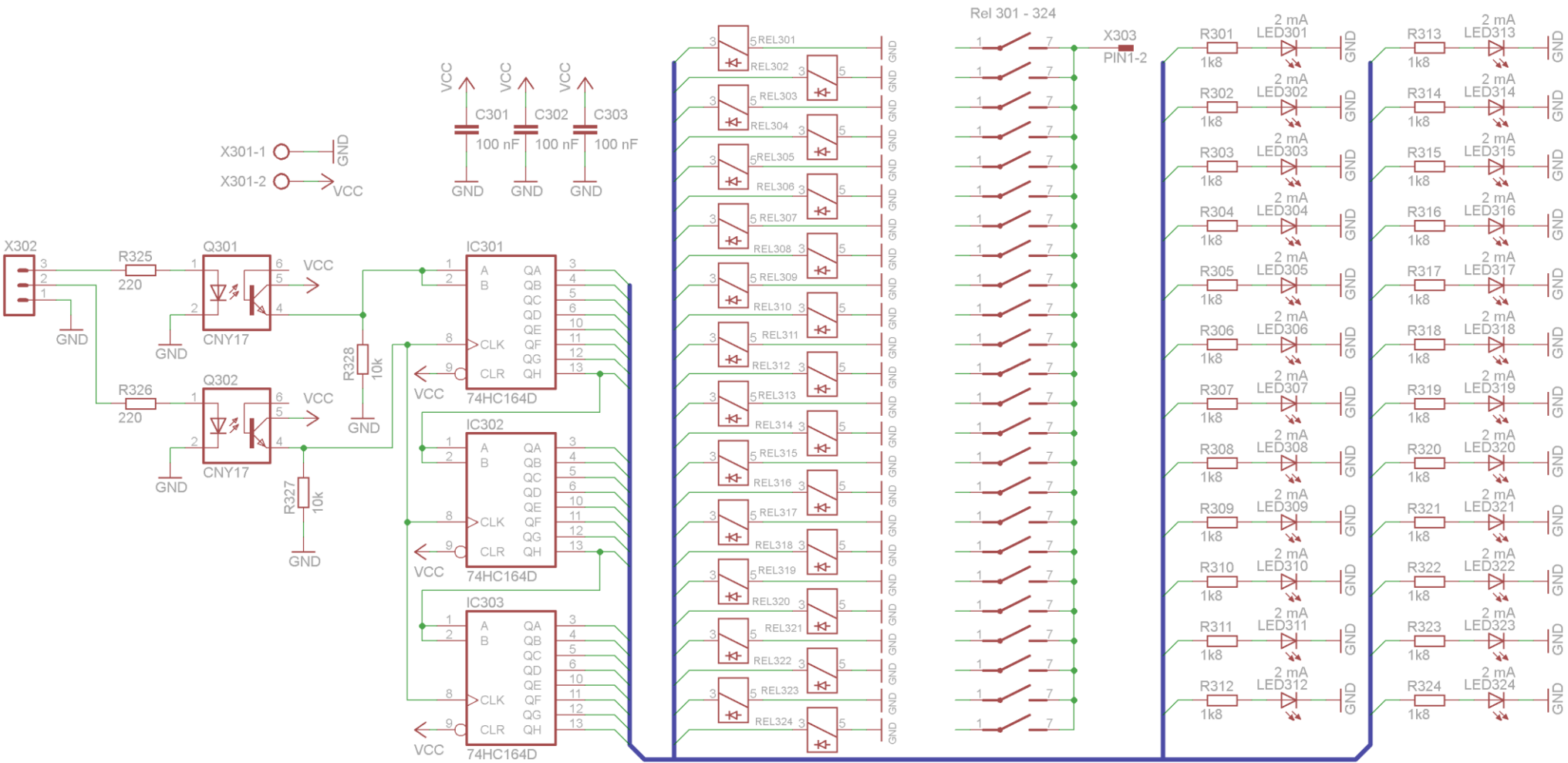


## Příloha B2

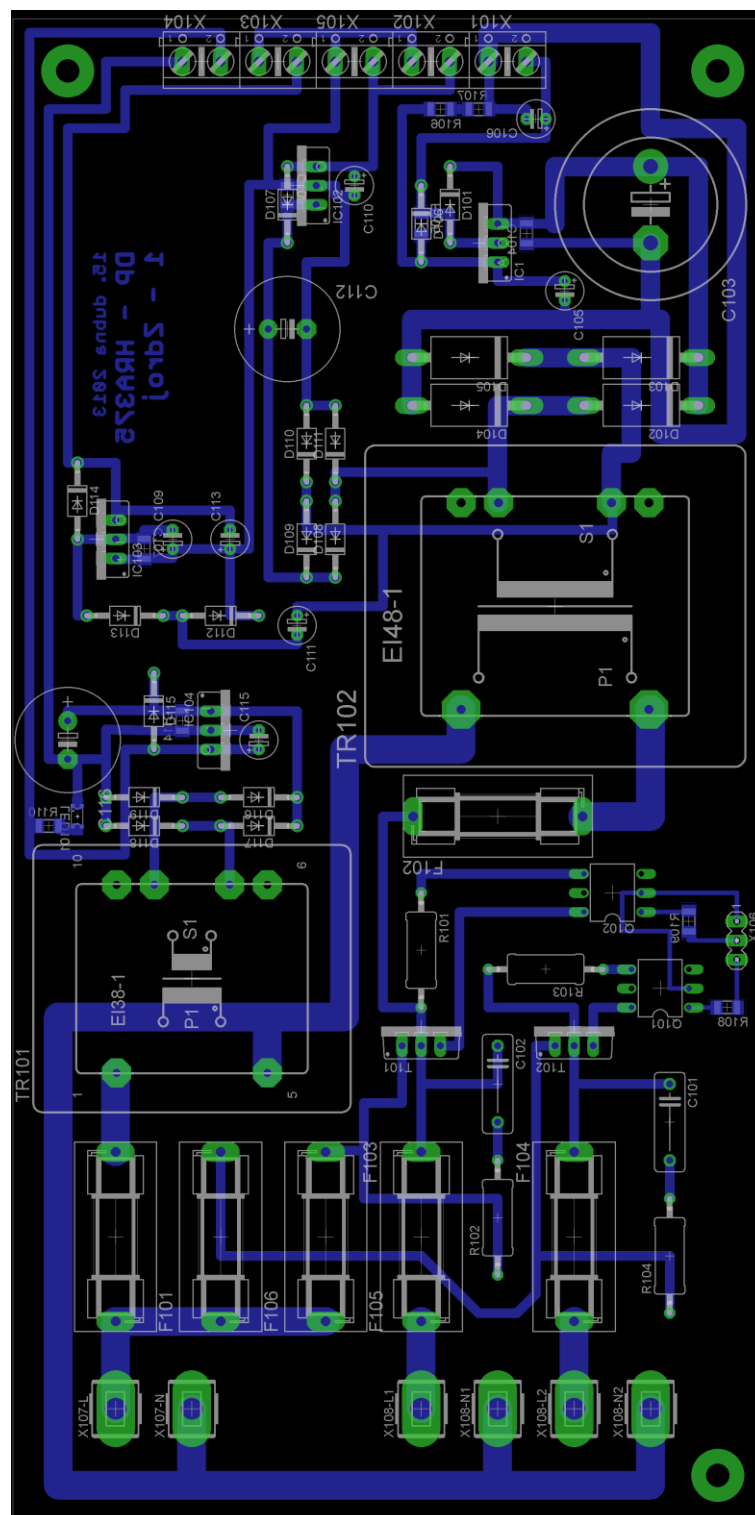




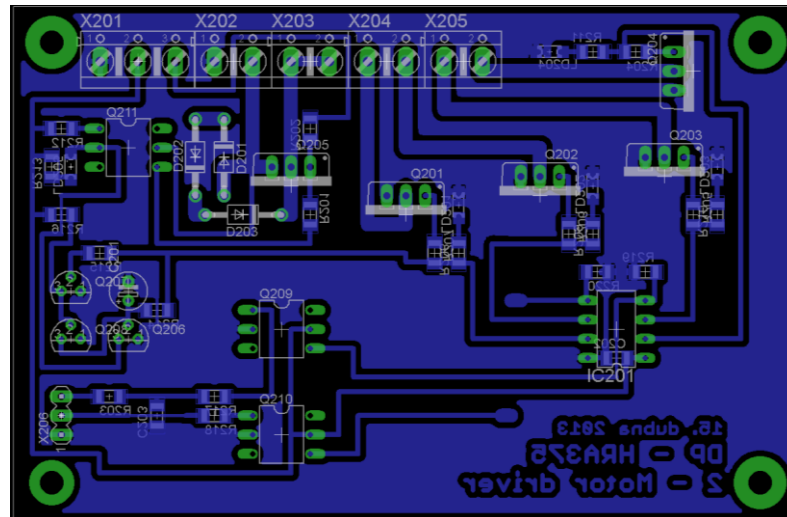
Příloha B3

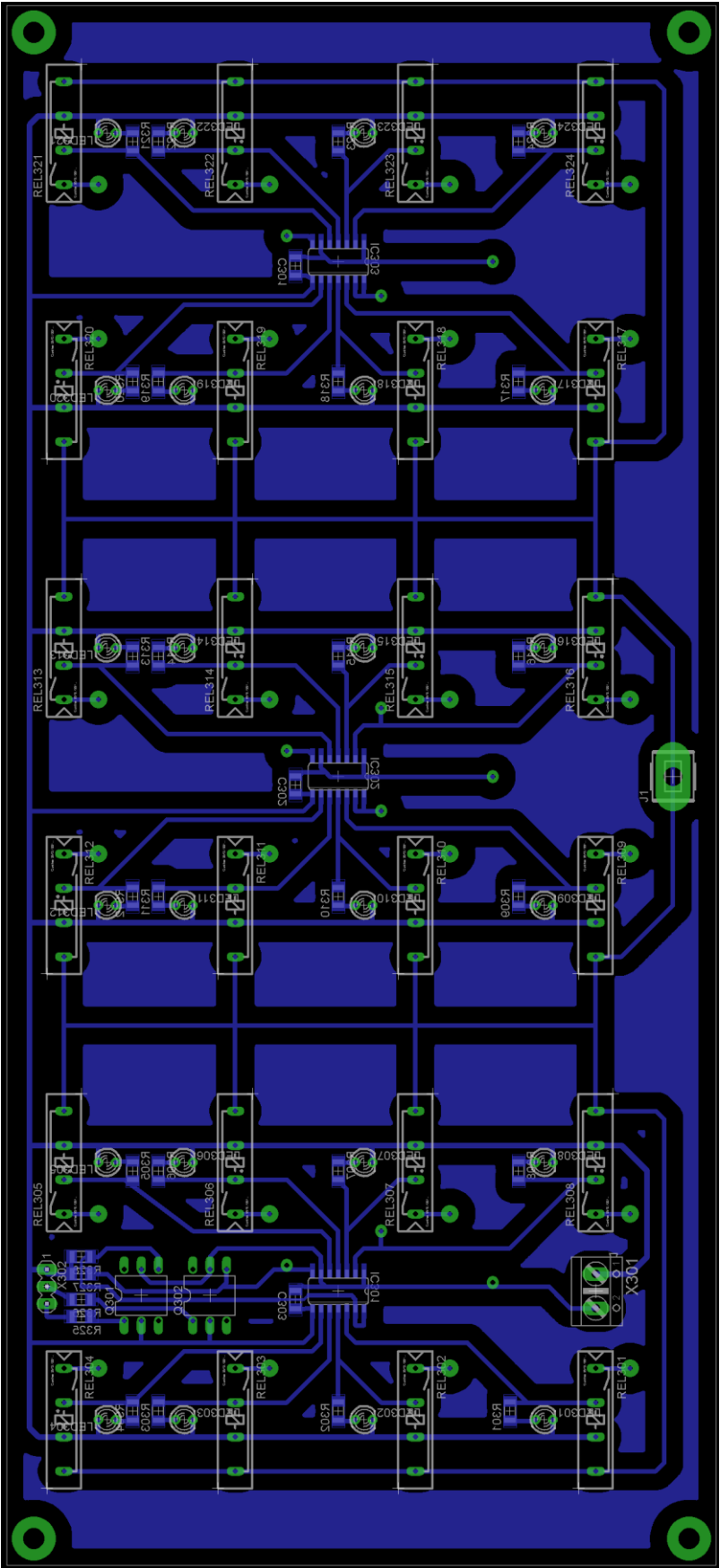


## Příloha C1

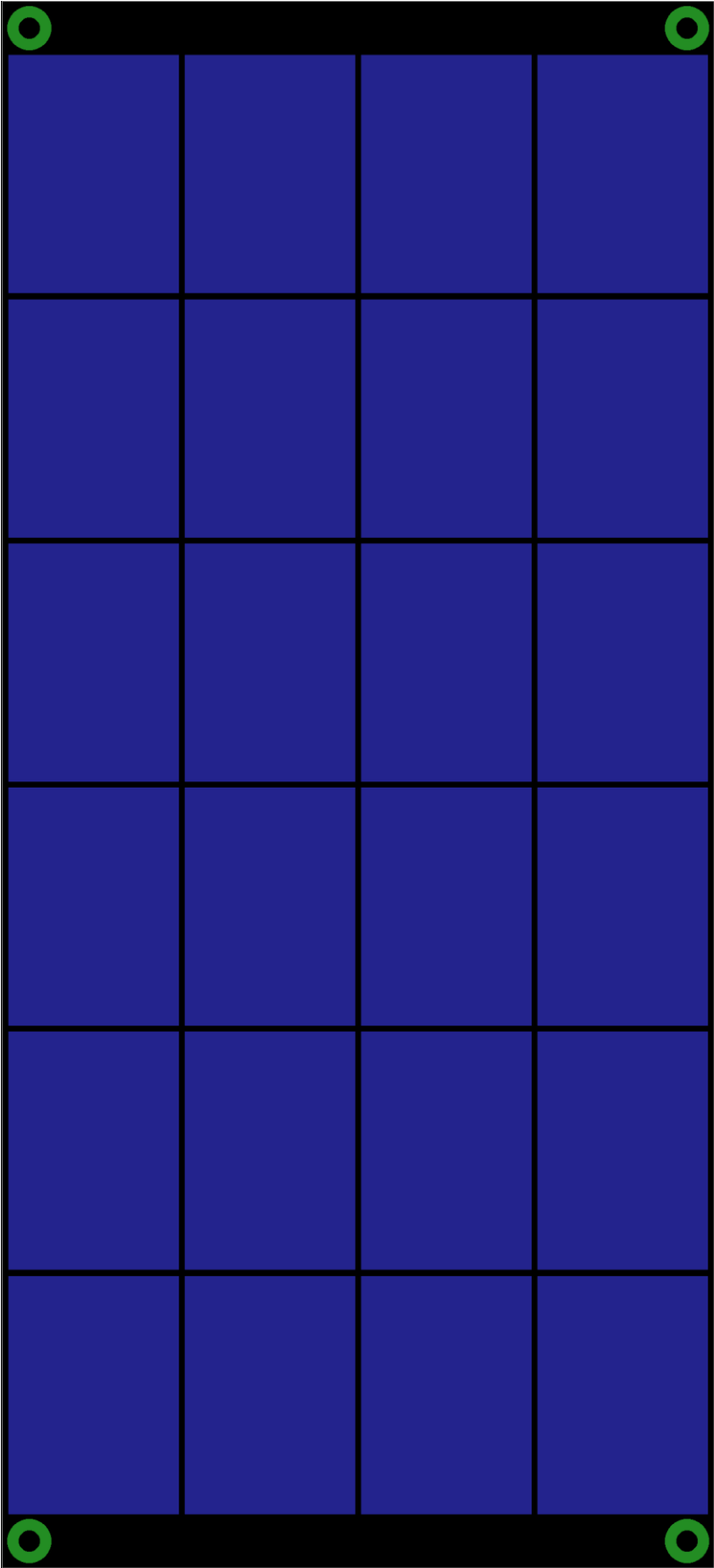


## Příloha C2

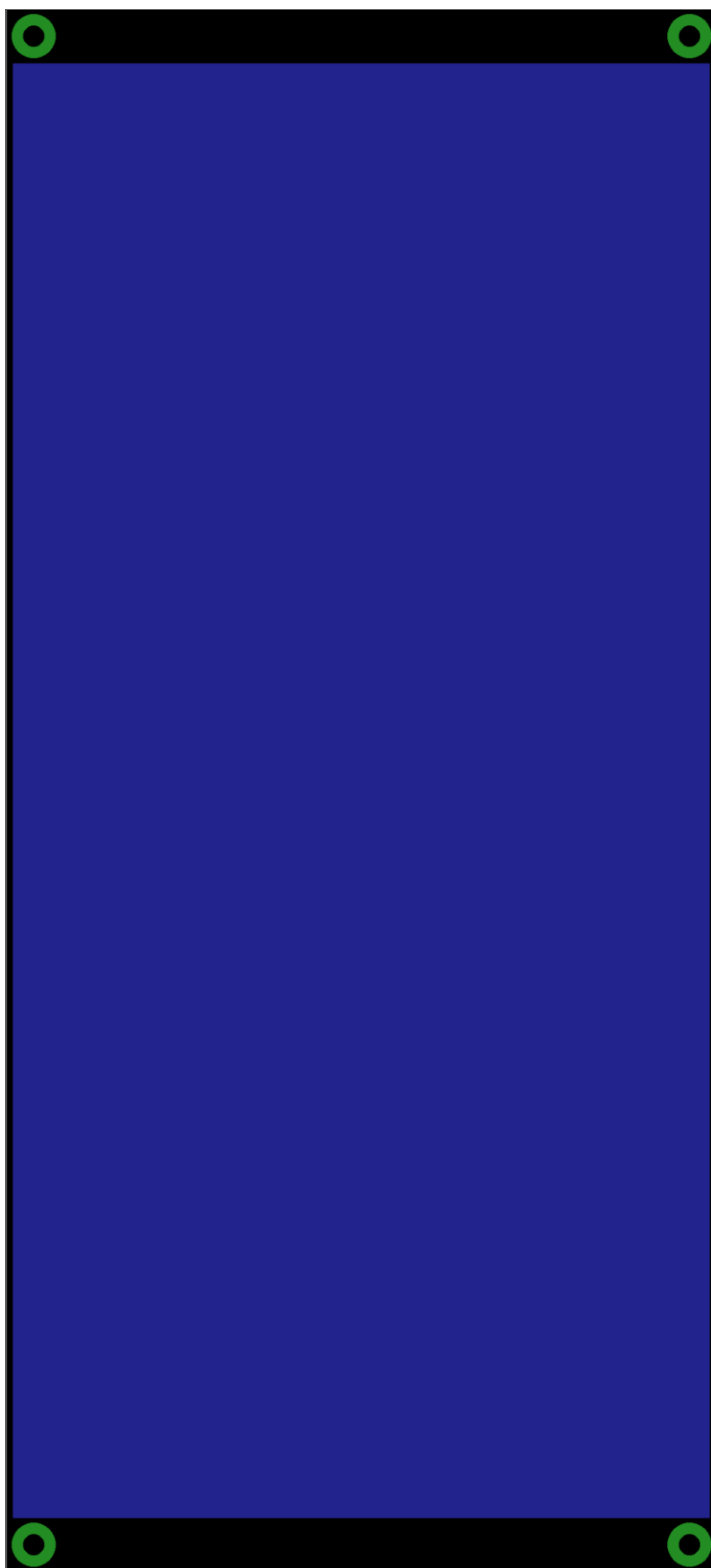




Příloha C4



**Příloha C5**



## Příloha D

Obsah jednotlivých adresářů na DVD s touto prací:

- ***Aplikace pro Raspberry Pi*** – aplikace pro Raspberry Pi ve formátu projektu vývojového prostředí MonoDevelop.
- ***Diplomová práce*** – tento dokument.
- ***Katalogové listy*** – katalogové listy, datasheety, všech důležitých použitých součástí.
- ***Laboratorní úloha*** – podklady ke stávající laboratorní úloze „Kapacitní snímače“.
- ***MCU*** – program pro mikrokontrolér v JSA a v HEX souboru.
- ***Obraz paměťové karty*** – obraz paměťové karty pro Raspberry Pi.
- ***Schémata, DPS*** – schémata zapojení a nákresy desek plošných spojů ve formátu programu Eagle verze 6.2.0.
- ***Webová aplikace*** – soubory potřebné pro funkci webového uživatelského rozhraní na Raspberry Pi.